NOTICE: When government or other drawings, specifications or other data are used for any purpose other than in connection with a definitely related government procurement operation, the U. S. Government thereby incurs no responsibility, nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may in any way be related thereto.

63-2-5

NRL Report 5837

A DIGITAL COMPUTER PROGRAM FOR CALCULATING SPHEROIDAL WAVE FUNCTION EIGENVALUES AND EXPANSION CONSTANTS

S. Hanish and R. V. Baier

Transducer Branch
Sound Division

January 31, 1963

U. S. NAVAL RESEARCH LABORATORY
Washington, D.C.

CONTENTS

ABSTRACT

Using the well-known procedures of Bouwkamp ("Theoretische en numerieke behandling van de buiging door een ronde opening," Diss. Groningen, Groningen Batavia, 1941) for finding the eigenvalues of differential equations solvable by application of three-term recursion formulas, the authors have developed a digital computer program in symbolic language for obtaining the eigenvalues of the separated steady-state wave equation in spheroidal coordinates, i.e., for the equation

$$\frac{d}{d\eta}\left[(1-\eta^2)\frac{dS_{m\ell}}{d\eta}\right] + \left[A_{m\ell} - h^2\eta^2 - \frac{m^2}{1-\eta^2}\right] S_{m\ell} = 0$$

where the parameters ℓ, m, and h are as defined by Stratton et al. in "Spheroidal Wave Functions" (New York:Wiley, 1956). The symbolic language used is universal (e.g., add a to b, multiply c by d, etc.) and may be applied to any digital computer by simple conversion to machine language. The program developed requires at least 3400 working cells. It computes the eigenvalue $A_{m\ell}$ to a maximum of 13 significant places for any integer values of ℓ and ($\ell \geq m$). The value of h is left arbitrary. The program also computes all the spheroidal wave function expansion coefficients (d_p) up to $p \leq 200$.

PROBLEM STATUS

This is an interim report on one phase of the problem; work is continuing.

AUTHORIZATION

NRL Problem S02-07
Project RF 001-03-44-4052

Manuscript submitted August 15, 1962.

A DIGITAL COMPUTER PROGRAM FOR CALCULATING SPHEROIDAL WAVE FUNCTION EXPANSION CONSTANTS

INTRODUCTION

The solution of the Helmholtz wave equation in spheroidal coordinates leads to spheroidal wave functions of the angle and radial types. It is customary to expand these wave functions in two infinite series consisting of associated Legendre functions and spherical Bessel functions, respectively. Tables of expansion coefficients have appeared in the literature at various times. Unfortunately, these tables have not been unique since the entries depended upon the definitions of the functions as conceived by their authors and upon the author's choice of normalization. In the decade 1950-1960 there appeared, among others, two new books of tables, one by Stratton, Morse, et al. (1) and the other by Flammer (2). The first collection lists the expansion coefficients $d_n\,(h|m\ell)$ for $h = 0.1$ (0.1) 1, $h = 1$ (0.2) 8, $m = 0$ (1) 8, and $\ell = 0$ (1) 8. The second book is a miscellany from many sources covering some of the ranges in the first but normalized by an entirely different scheme. Flammer's book, in addition, contains a cogent account of wave function theory. The extent of these existing tables has been found to be insufficient for many computation needs in the field of applied acoustics. The range of the eccentricity parameter h is too restrictive, and interpolation for h values not listed is impractical.

The authors of the present report have therefore turned to the difficult task of writing a digital computer program which is universal enough to supply spheroidal wave function expansion constants in any quantity for any values of m, n, and h. Our efforts have been considerably lightened by the lucid explanation in Flammer's book, for which we acknowledge a debt of gratitude. Sections of Flammer's work appear in the following brief explanation of the procedure we used to formulate the computer program.

ANGLE SPHEROIDAL WAVE FUNCTIONS

Prolate spheroidal angle functions $S(h,\eta)$ are defined to be functions which satisfy the ordinary differential equation

$$\frac{d}{d\eta}\left[(1-\eta^2)\frac{dS}{d\eta}\right] + \left[A - h^2\eta^2 - \frac{m^2}{1-\eta^2}\right] S = 0. \qquad (1)$$

The parameters A and m are separation constants. We specify here that m is an integer and that $A(h) \to \ell(\ell+1)$ as $h \to 0$, where ℓ is the order (an integer) of the associated Legendre function $P_\ell^m(\eta)$. For each value of the integers m and ℓ there corresponds a function $S_{m\ell}$ which can be expanded in an infinite series of associated Legendre functions. The function $S_{m\ell}$ is

$$S_{m\ell}\,(h,\eta) = \sum_n{}' \; d_n\,(h|m\ell)\, P_{m+n}^m\,(\eta). \qquad (2)$$

If one substitutes Eq. (2) into Eq. (1), one obtains the following three-term recursion formula in $d_n(h|m\ell)$;

$$\rho_{m,n}\, d_{n+2} + \sigma_{m,n}\, d_{n-2} + (\tau_{m,n} - A_{m\ell})\, d_n = 0 \qquad (3a)$$

where

$$\rho_{m,n} = \frac{(n + 2m + 1)\,(n + 2m + 2)}{(2n + 2m + 3)\,(2n + 2m + 5)}\, h^2, \tag{3b}$$

$$\sigma_{m,n} = \frac{n(n - 1)}{(2m + 2n - 1)\,(2m + 2n - 3)}, \tag{3c}$$

and

$$\tau_{m,n} = \left[\frac{2\,(n + m)\,(n + m - 1) - 2m^2 - 1}{(2n + 2m + 3)\,(2n + 2m - 1)}\right] h^2 + (n + m)\,(n + m + 1). \tag{3d}$$

A simple rearrangement of Eq. (3a) gives

$$\frac{d_n}{d_{n-2}} = \frac{-\sigma_{m,n}}{\tau_{m,n} - A_{m\ell} + \rho_{m,n}\dfrac{d_{n+2}}{d_n}}. \tag{4a}$$

If one multiplies both sides of Eq. (4a) by $\rho_{m,n-2}$ one obtains

$$N_n^m = \frac{-\beta_n^m}{\tau_n^m - A_{m\ell} + N_{n+2}^m} \tag{4b}$$

or

$$\left[N_{n+2}^m\right]_1 = \frac{-\beta_{n+2}^m}{\tau_{n+2}^m - A_{m\ell} + N_{n+4}^m} \tag{4c}$$

in which

$$N_n^m = \rho_{m,n-2}\,\frac{d_n}{d_{n-2}}, \quad N_{n+2}^m = \rho_{m,n}\,\frac{d_{n+2}}{d_n},$$

and

$$\beta_n^m = \rho_{m,n-2}\,\sigma_{m,n}\,.$$

One can also solve Eq. (4b) for N_{n+2}^m and obtain

$$\left[N_{n+2}^m\right]_2 = \frac{-\beta_n^m}{N_n^m} + A_{m\ell} - \tau_n^m\,. \tag{4d}$$

Equations (4c) and (4d) are continued fractions. If one subtracts these equations for a given $A_{m\ell}$ and the result is zero, the number $A_{m\ell}$ is a characteristic value of Eq. (1). In symbols, to determine $A_{m\ell}$ we attempt to make the following calculation

$$\left[N_{n+2}^m\right]_1 - \left[N_{n+2}^m\right]_2 \cong 0. \tag{4e}$$

The first term on the left-hand side is computed by initially taking n so large that N_{n+4}^m is negligible, and then finding the continued fraction $[N_{n+2}^m]_1$ for successively smaller values of n. The second term on the left-hand side is computed by assuming N_{n+2}^m is zero for $n+2 < 0$ and then computing $[N_{n+2}^m]_2$ for successively larger values of n. It is evident from this procedure that one can obtain $A_{m\ell}$ only by successive approximation. A very effective method of hastening the computation of $A_{m\ell}$ has been devised by Bouwkamp (3).

In this method one first computes $A_{m\ell}$ by allowing $A_{m\ell} = \ell(\ell + 1)$. Then one computes a correction term $\delta(A_{m\ell})$ by taking the variation of Eq. (4e) due to a variation $\delta(A_{m\ell})$; in symbols this is

$$\left[N^m_{n+2}\,(A_{m\ell} + \delta(A_{m\ell}))\right]_1 - \left[N^m_{n+2}\,(A_{m\ell} + \delta(A_{m\ell}))\right]_2 = 0. \tag{5a}$$

Now

$$N^m_{n+2}\,(A_{m\ell} + \delta(A_{m\ell})) \approx N^m_{n+2}\,(A_{m\ell}) + \delta(N^m_{n+2}). \tag{5b}$$

Hence, Eq. (5a) reads

$$\left[N^m_{n+2}\,(A_{m\ell})\right]_1 - \left[N^m_{n+2}\,(A_{m\ell})\right]_2 + \delta\left[N^m_{n+2}\right]_1 + \delta\left[N^m_{n+2}\right]_2 = 0. \tag{5c}$$

From Eq. (4c) one finds

$$\delta\left[N^m_{n+2}\right]_1 = \frac{\beta^m_{n+2}\left[-\delta(A_{m\ell}) + \delta(N^m_{n+4})\right]}{(\tau^m_{n+2} - A_{m\ell} + N^m_{n+4})^2}$$

$$= -\delta(A_{m\ell})\left[\frac{(N^m_{n+2})^2}{\beta^m_{n+2}} + \frac{(N^m_{n+2})^2\,(N^m_{n+4})^2}{\beta^m_{n+2}\;\beta^m_{n+4}} + \cdots\right]. \tag{6a}$$

Similarly, from Eq. (4d)

$$\delta\left[N^m_{n+2}\right]_2 = \frac{\beta^m_n}{(N^m_n)^2}\,\delta(N^m_n) + \delta(A_{m\ell})$$

$$= \delta(A_{m\ell})\left[1 + \frac{\beta^m_n}{(N^m_n)^2} + \frac{\beta^m_{n-2}\,\beta^m_n}{(N^m_n)^2\,(N^m_{n-2})^2} + \cdots\right]. \tag{6b}$$

Since n is an integer it is convenient to let $n = \ell - m$. Substituting Eqs. (6a) and (6b) into Eq. (5c) and solving for $\delta(A_{m\ell})$ one obtains

$$\delta(A_{m\ell}) = \frac{\left[N^m_{\ell-m+2}\right]_1 - \left[N^m_{\ell-m+2}\right]_2}{\left[1 + \dfrac{\beta^m_{\ell-m}}{(N^m_{\ell-m})^2} + \dfrac{\beta^m_{\ell-m-2}\,\beta^m_{\ell-m}}{(N^m_{\ell-m-2})^2\,(N^m_{\ell-m})^2} + \cdots\right] + \left[\dfrac{(N^m_{\ell-m+2})^2}{\beta^m_{\ell-m+2}} + \dfrac{(N^m_{\ell-m+2})^2\,(N^m_{\ell-n+4})^2}{\beta^m_{\ell-m+2}\;\beta^m_{\ell-m+4}} + \cdots\right]}. \tag{6c}$$

Repeated corrections by Eqs. (6c) and (5a) lead to very accurate values of the characteristic number $A_{m\ell}$. Now

$$\frac{d_n}{d_{n-2}} = \frac{N^m_n}{\rho_{m,n-2}}\,. \tag{7a}$$

The number N^m_n is calculated from Eq. (4b) using the corrected value of $A_{m\ell}$. The ratio d_n/d_{n-2} is therefore accurately calculated. From Eq. (7a) one deduces that

$$d_n = d_0 \left[\frac{d_n}{d_{n-2}} \frac{d_{n-2}}{d_{n-4}} \cdots \frac{d_2}{d_0}\right],\ n \geq 2. \tag{7b}$$

The expansion constants d_n can be normalized in several different ways. We adopt here the normalization of Morse and Feshbach (4),

$$\sum_{n=0,1}' \frac{(n+2m)!}{n!} d_n = \frac{(\ell+m)!}{(\ell-m)!}. \tag{7c}$$

Hence, the magnitude of d_0 is fixed to be

$$d_0 = \frac{(\ell+m)!}{(\ell-m)!\left\{\sum_{n=2}^{\infty}{}' \frac{(n+2m)!}{n!}\left[\frac{d_n}{d_{n-2}} \frac{d_{n-2}}{d_{n-4}} \cdots \frac{d_2}{d_0}\right] + (2m)!\right\}} \text{ for n even.} \tag{7d}$$

Similarly we find d_1 to be

$$d_1 = \frac{(\ell+m)!}{(\ell-m)!\left\{\sum_{n=3}' \frac{(n+2m)!}{n!}\left[\frac{d_n}{d_{n-2}} \frac{d_{n-2}}{d_{n-4}} \cdots \frac{d_3}{d_1}\right] + (1+2m)!\right\}} \text{ for n odd.} \tag{7e}$$

All remaining d_n are found by substituting Eqs. (7d) and (7e) into Eq. (7a).

The prolate spheroidal angle functions $S(h,\eta)$ can be converted into oblate spheroidal angle functions by making the transformation $h \to -jg$. The prolate expansion coefficients $d_n(h|m\ell)$ become $d_n(-jg|m\ell)$ by this same transformation. Thus to find $d_n(-jg|m\ell)$ one replaces h^2 in Eq. (3a), (3b), (3c), and (3d) by $-g^2$. The normalization procedure is identical for both prolate and oblate cases.

THE COMPUTER PROGRAM

The computer program is divided into two parts. In the first part the program computes the eigenvalue $A_{m\ell}$ to 13 significant places, or less depending on the parameters m, ℓ, h. In the second part the program computes any number of spheroidal wave function expansion constants $d_n(h\,|\,m\ell)$ where n may take on integer values up to $n = 400$. These constants are computed to 13 significant places or less depending on the parameters m, ℓ, h. The precision of $A_{m\ell}$ controls the precision of $d_n(h/m\ell)$.

The computation of eigenvalues is accomplished by Bouwkamp's procedure (3), the essential step of which is found in Eq. (6c). We select a starting value of $A_{m\ell}$ according to the given magnitude of h. A short discussion of the method of selecting a good starting value for $A_{m\ell}$ is presented in Appendix E of this report. If this starting value is poor the program may converge to an erroneous eigenvalue. After the initial value of $A_{m\ell}$ is set, the value is improved by application of Eq. (6c). Repeated use of Eq. (6c) eventually leads to very accurate values of $A_{m\ell}$.

When the computation of the eigenvalue is completed the computation of the constants d_n is accomplished by using the relations found in Eqs. (7a) to (7e).

The computer program finally assembled is reproduced in Appendix B of this report. The meanings of the various instructions used in the program may be found in the publication Narec Bulletin No. 27 (July 27, 1961) available from the Research Computation Center at NRL. We note that the language of this program is entirely symbolic (i.e., add a to b, multiply c by d, etc.). The language is therefore universal. It may be converted to the machine language of a specific computer by replacing each symbolic command by the equivalent machine language command.

A set of input instructions for the particular version of this symbolic program in the NIP language of the Narec-NRL is found in Appendix A.

ADDITIONAL REMARKS

The authors of this report are in the process (to date) of preparing digital computer programs for computing radial and angle spheroidal wave functions of any order and any argument, using the eigenvalues and expansion coefficients found by the program described here.

ACKNOWLEDGMENT

We are grateful to Mr. M. Brinkman of the Research Computation Center at NRL for the expert detailed assistance he gave us at all stages of our work on this involved program.

REFERENCES

1. Stratton, J.A., Morse, P.M., et. al., "Spheroidal Wave Functions," New York:Wiley, 1956

2. Flammer, C., "Spheroidal Wave Functions," Stanford Univ. Press., Stanford, Calif., 1957

3. Bouwkamp, C.J., "Theoretische en numerieke behandling van de buiging door een ronde opening," Diss. Groningen, Groningen-Batavia, 1941

4. Morse, P.M., and Feshback, H., "Methods of Theoretical Physics," New York:McGraw-Hill, 1953

5. Meixner, J., "Asymptotische Entwicklung der Eigenwerte und Eigenfunktionen der Differentialgleichungen der Sphäroid-Funktionen und der Mathieuschen Funktionen," Z. angew. math. Mech. 28:304-310 (1948)

APPENDIX A

INSTRUCTIONS FOR USE OF NAREC-NRL SPHEROIDAL WAVE FUNCTION ROUTINE

The Narec-NRL spheroidal wave function routine is found on Narec Tape No. 4593

1. Prepare a machine-language Operator Instruction Sheet (see Fig. A1).

2. Prepare a Data Input Sheet as follows:

a. Insert the prolate spheroidal values of m, ℓ, and h in successive two-word entries on the sheet, reserving two twelve-digit Narec floating point words (i.e., fractional part and exponent) for each of the three numbers. These values will take up six lines.

b. Determine the highest subscript (XXX) in the d_{XXX} constants desired. If ℓ-m is even, the subscript will be even; if ℓ-m is odd, the subscript will be odd. Insert the value of XXX as a two-line Narec floating point number in the two lines following the six noted above. This makes eight lines.

c. Count the actual number of d's to be typed out. If ℓ-m is even, the count is (XXX +2)/2. If ℓ-m is odd, the count is (XXX + 1)/2. Convert this count to a hex number and pre-fix zeros to make up a "4-digit hex number" (say YYYY). Insert this hex number in the standard word appearing in the ninth line. This word is

YYYY de 064a e5.

d. If ℓ-m is even, convert XXX + 2 into a four-digit hex number, add eight zeros, and insert in line 10.

If ℓ-m is odd, convert XXX + 1 into a four-digit hex number, add eight zeros, and insert in line 10.

When this procedure is complete the data tape contains 10 lines, forming one block of data input. At the head of the block write "nnnn" and at the tail write "mmmm."

Any number of blocks may be inserted, provided the separation symbols (nnnn and mmmm) are used. When the data input sheet is completed, it is handed to Narec Research Computation Center for further processing.

3. Manual Changes

a. If oblate spheroidal constants are desired, write on the blue machine-language Instruction Sheet

Manual Changes: line 0413 – 0587 e7 0587 eb

07b3 – 0800 00 0000 00

07b4 – 0001 00 0000 00

OPERATOR INSTRUCTION SHEET Date ____________

RCC Problem Number ____________ Estimated Computation Time 15 MIN

Problem Title ____________ Programmer ____________

Run ____________ Tel. Ext. ____________

OUTPUT FORMAT: Copies ______ original ______ carbons.
A: ~~Hexadecimal~~, Decimal: 2 cols. B: Hexadecimal, Decimal: --- cols.

INPUT TAPE INFORMATION

	Tape No.	Locations From	Locations Thru	Check Sum (if known)		Tape No.	Locations From	Locations Thru	Check Sum (if known)
1	4593	0400	0864	1875 A0 120240	6				
2	2700	0000	0344		7				
3	Put data tape on [illegible]				8				
4					9				
5					10				

SPECIAL INPUT INSTRUCTIONS, MANUAL CHANGES, AND STARTING ORDER

[illegible]

loc 0413 0587 C7 05 [illegible]

07[illegible]3 02000 [illegible]

07[illegible]4 00 1000 000000

PRINT-OUTS
Estimated time to First Print-out ______ Total Number of Output Lines ______

Time Per Line ______ Location of Final Stop Order ______

SAMPLE

RESTART ORDERS
Read in Tapes ______

Manual Changes ______

Starting Order LO RO ______

DUMP: Output Format A B Other ______
Read in Tapes ______

Manual Changes ______

Starting Order LO RO ______

SPECIAL INSTRUCTIONS, ADDITIONAL INFORMATION, ETC.

Fig. A1 - Sample of Operator Instruction Sheet

b. If the tolerance on $A_{m\ell}$ is to be changed, compute the new tolerance as a two-line Narec floating point number and write this number as

Manual Change: line 0610 ____ __ ____ __

0611 ____ __ ____ __

The tolerance on $A_{m\ell}$ is fixed at 10^{-11} in the program.

c. If the tolerance used in comparing $N^m_{\ell-m+2}$ with $-\gamma^m_n + A_{m\ell}$ is to be changed, compute a new tolerance as a two-line Narec floating point number and write

Manual Change: line 05fe: ____ __ ____ __

05ff: ____ __ ____ __

This tolerance is started at 10^{-2} in the program.

d. If the eigenvalue tolerance (= ℓ tol) comparing the change computed eigenvalue for increasing N's is to be changed, compute a new tolerance as a two-line Narec floating point number and write

Manual Change: line 06d0: ____ __ ____ __

06d1: ____ __ ____ __

This tolerance is fixed at 10^{-11} in the program.

e. If the nth ratio tolerance (= n tol) comparing the change in $N^m_{\ell-m+4}$ as the number of N's increase is to be changed, compute a new tolerance as a two-line Narec floating point number and write

Manual Change: line 06C3: ____ __ ____ __

06C4: ____ __ ____ __

This tolerance is fixed at 10^{-11} in the program.

f. Starting values for $A_{m\ell}$ depend upon whether h is considered "small" or "large." A short discussion of the method of making a proper selection of the starting value for $A_{m\ell}$ is found in Appendix E.

As an example of the use of the instructions outlined above, we select the following as an appropriate example of the preparation of a data input sheet.

Example: Let $m = 1$, $\ell = 7$, and $h = 10$ for the oblate case. We desire the largest expansion constant to be d_{60} (note that the subscript must be an even integer). The data input sheet reads as follows.

n n n n n

080000 000000

000100 000000

070000 000000

000400 000000

0A0000 000000

000400 000000

0f0000 000000

000600 000000

001fde 064Ae5

003E00 000000

m m m m m

This sheet is sent to Narec Research Computation Center for further processing.

Since the spheroid is oblate, make the following "manual changes" on the standard blue-colored Narec Operator Instruction Sheet:

line 0413 → 0587e7 0587eb

07b3 → 080000 000000

07b4 → 000100 000000

APPENDIX B

REPRODUCTION OF COMPUTER PROGRAM IN SYMBOLIC FORM

NOTE: Any group of letters and numbers not specifically appearing in the list of commands of Narec Bulletin #27 is to be considered a machine address. For example, spro, kl, k2, etc. are addresses, while p, pd, mp are standard commands.\ Method of reading: at address spro in the computer memory place the command "go to left half of address bgn etc.

```
spro    lo bgn
        0
kl      kl a
        ro 512
k2      j3 p
        -j2 pd
        p j9
        j3 p
        j5 pd
        p j10
        j3 p
        mp j6
        p j11
        -j5 pd
        p j12
        j8 pd
        p j13
        j2 p
        mp j2
        mp j6
        p j14
        mp j6
        p j15
        j2 p
        j5 pd
        p j16
        j2 p
        mp j6
        p j17
        -j5 pd
        p j18
        j6 pd
        p j19
        -j8 pd
        p j20
        j8 pd
        j6 pd
        p j21
        j4 p
        mp j4
        p j22
        p/ j6
        p j23
```

```
        j22 p
        mp j22
        p j24
        j3 p
        mp j10
        p j25
        j11 p
        mp j10
        -j14 pd
        -j5 pd
        p j26
        j12 p
        mp j13
        p j27
        j26 p
        p/ j27
        mp j22
        j25 pd
        kfdg pd
        p j28
        fl yadl
k3      j9 p
        p j100
k4      j100 p
        cmp j5
sn7     fl k5
        fl k6
sn8     fl k7
k7      j100 p
        -j6 pd
        p j100
        fl k4
k5      jq20 p
        p jq30
        p jq31
        fl k12
k6      j5 p
        p jq30
        p jq31
        fl k12
k8      j2 p
        jq31 pd
        p j32
        j5 pd
        mp j32
        p j33
        j15 p
```

```
        -j5 pd
        p j34
        jq31 p
        mp j6
        p j35
        j18 p
        j35 pd
        p j36
        j8 pd
        mp j36
        p j37
        j34 p
        p/ j37
        p j38
        j5 p
        -j38 pd
        mp j23
        j33 pd
        p j39
k9      fr 0
k10     jq31 p
        -j5 pd
        p j40
        j17 p
        jq31 pd
        p j41
        -j5 pd
        p j42
        jq31 pd
        p j45
        j6 pd
        p j44
        -j8 pd
        p j43
        jq31 p
        mp j40
        mp j41
        mp j42
        p/ j45
        p/ j45
        p/ j43
        p/ j44
        mp j24
        p j101
k11     fr 0
k12     j5 p
        p j900
        j9 p
        j8 pd
khb     p j102
        jq30 p
        p j103
        p jq31
k13     j103 p
        j6 pd
sn1     p j103
        cmp j102
sn9     fl kg14
        fl k19
sna     fl k19
kg14    nl k14
k14     k16 a
        lo k15
k15     lal k11
        il k11
k16     k16 a
        lo 512
k17     fl k10
        j101 p
k18     p j400
        adl 2
        k18
        jq31 p
        j6 pd
        p jq31
        fl k13
k19     jq30 p
        p jq31
        p j104
kg19    j104 p
        j6 pd
sn2     p j104
        cmp j102
snb     fl qq1
        fl kg24
snc     fl kg24
qq1     nl k20
k20     k22 a
        lo k21
k21     lal k9
        il k9
k22     k22 a
        lo 512
        fl k8
        j39 p
k23     p j300
        adl 2
        k23
        jq31 p
        j6 pd
        p jq31
        fl kg19
kg24    jq30 p
        p jq31
        p j105
k24     j105 p
        j6 pd
sn3     p j105
        cmp j102
snd     fl k25
gk24    fl hv10
        fl hv10
k25     -j400 p
k26     p/ j900
k27     -j300 pd
        j28 pd
k28     p j901
        p j106
        adl 2
        k25
        adl 2
        k26
        adl 2
        k27
        adl 2
        k28
        fl k24
hv10    j9 p
        j6 pd
        p j120
```

```
        p j123
        -j6 pd
        p j124
        jq30 p
        p j121
hv11    j121 p
        j6 pd
sn4     p j121
        cmp j123
sne     fl hv12
        fl hv15
snf     fl hv15
hv12    j401 p
hv13    p/ j901
wb10    p/ j901
hv14    p v100
        adl 2
        hv12
        adl 2
        hv13
        adl 2
        hv14
        adl 2
        wb10
        fl hv11
hv15    lsl hv14
        hv18
hv17    j5 p
hv18    p 0
hv22    v100 p
hw21    cmp j5
        fl hw22
sng     fl hv28
        fl hw22
hw22    lsl hv22
        hv23
hu22    lsl hv22
        hw24
hw23    adl 2
        hv23
hw24    0 p
        p w100
        jq30 p
        p j122
hw25    w100 p
hv23    mp 0
hv24    p w100
hv25    j122 p
        j6 pd
sn5     p j122
        cmp j124
snh     fl hv26
        fl hv27
sni     fl hv27
hv26    adl 2
        hv23
        fl hw25
hv27    adl 2
        hv22
        adl 2
        hv24
        adr 2
        hw24
        adl 2
        hw25
```

```
        j124 p
        -j6 pd
        p j124
        fl hv22
j124    0
        0
        0
        0
hv28    lsr hv24
        hv29
hv29    j5 p
        p 0
hv30    w100 p
        cmp j5
snj     fl hv31
        fl hv33
snk     fl hv31
hv31    yz12 p
hv32    w100 pd
        p yz12
        adl 2
        hv30
        adl 2
        hv32
        fl hv30
hv33    yz12 p
        j5 pd
        p yz12
k29     j9 p
        p j107
k30     j107 p
        j6 pd
        p j107
        p j111
        p j112
        p jq31
        nl k31
k31     k33 a
        lo k32
k32     lal k11
        il k11
k33     k33 a
        lo 512
k34     fl k10
        nl k35
k35     k37 a
        lo k36
k36     lal k9
        il k9
k37     k37 a
        lo 512
k38     fl k8
        j39 p
        -j28 pd
        p j108
        mp wz10
        /p/
        p j109
        -j101 p
sn6     p/ j108
        /p/
snm     cmp j109
        fl kg39
snn     fl kg39
        fl k30
```

```
kg39    jq30 p
        p jq31
k39     j9 p
        -j6 pd
        p j110
k40     j111 p
        -j6 pd
        p j111
k41     j110 p
        -j111 pd
        fcl k55
k42     j112 p
        p jq31
        nl k43
k43     k45 a
        lo k44
k44     lal k11
        il k11
k45     k45 a
        lo 512
k46     fl k10
        nl k47
k47     k49 a
        lo k48
k48     lal k9
        il k9
k49     k49 a
        lo 512
k50     fl k8
        j39 p
        -j28 pd
        j113 pd
        p j114
        -j101 p
        p/ j114
        p j113
k51     j113 p
        mp j113
        p/ j101
k52     p k300
k53     j112 p
        -j6 pd
        p j112
k54     adl 2
        k52
        fl k40
k55     lsr k52
        k56
k56     j5 p
        p 0
hv56    j5 p
        p yz10
hv57    k300 p
        cmp j5
sno     fl k58
        fl hv64
snp     fl k58
k58     k300 p
        cmp j5
snq     fl k59
        fl kv63
snr     fl k59
k59     yz10 p
k60     mp k300
k61     p y100
        p yz10
        adl 2
        k58
        adl 2
        k60
        fl k58
kv63    adl 2
        hv57
        lsl hv57
        k58
        lsl hv57
        k60
        adl 2
        k61
        fl hv56
hv64    lsr k61
        kv65
kv65    j5 p
        p 0
k64     y100 p
        cmp j5
sns     fl k65
        fl k67
snt     fl k65
k65     yz11 p
k66     y100 pd
        p yz11
        adl 2
        k64
        adl 2
        k66
        fl k64
k67     yz12 p
        yz11 pd
        p yz13
        j113 p
        -j106 pd
        p/ yz13
        p yz14
        j28 pd
snu     p j28
        yz14 p
        /p/
        -j126 pd
k62     fcl k79
        fl ltcl
k70     pt 11
        fl hov 4
k68     ro k68
        lo dj8
k69     j151 p
        p j103
        p j104
        p j105
        p j121
        p j122
        p j123
        p yz11
        p yz12
        p j107
        p j111
        p j112
        p j113
        nl apne
dj8     dj8 a
```

```
        ro 512
dj9     j5 p
        p j900
dj10    dj50 p
        lsl dj10
        gk24
        j9 p
        j8 pd
        p j102
        j151 p
        p j103
        p j104
        p j105
        lsl w18
        k18
        lsl w23
        k23
        lsl w18
        k25
        lsl w9
        k26
        lsl w23
        k27
        lsl w91
        k28
        fr khb
dj11    lsr djf6
        dj12
dj12    j5 p
        p 0
        jq30 p
        p wy10
vh10    wy10 p
        j6 pd
        p wy10
vh11    j901 p
        /p/
hov6    cmp j5
        fl vh12
        fl vh16
        fl vh12
vh12    wy10 p
        mp j6
        j17 pd
        j5 pd
        p wy11
        -j6 pd
        mp wy11
        p wy12
        wy10 p
        j17 pd
        p wy13
        -j5 pd
        mp wy13
        p wy14
        wy12 p
        p/ wy14
        p/ j22
vh13    mp j901
vh14    p b100
vh15    adl 2
        vh11
        adl 2
        vh13
        adl 2
        vh14
        fl vh10
vh16    lsr vh14
        vh17
vh17    j5 p
        p 0
vh19    cx10 p
        j5 pd
        p cx10
vh20    b100 p
        /p/
        cmp j5
        fl vh21
        fl vh24
        fl vh21
vh21    cx10 p
vh22    mp b100
vh23    p c100
        p cx10
        adl 2
        vh20
        adl 2
        vh22
        adl 2
        vh23
        fl vh20
vh24    lsr vh23
        vh25
vh25    j5 p
        p 0
vh27    jq30 p
        p cq10
vh28    cq10 p
        j6 pd
        p cq10
        cq10 p
        j17 pd
        p qc10
        p qc12
vh29    c100 p
        /p/
        cmp j5
        fl vh30
        fl vh35
        fl vh30
vh30    cq10 p
        /p/
        cmp qc10
        fl vh31
        fl vh32
        fl vh32
vh31    qc10 p
        -j5 pd
        p qc11
        mp qc12
        p qc12
        qc11 p
        p qc10
        fl vh30
vh32    qc12 p
        p/cq10
vh33    mp c100
vh34    p vx10
        adl 2
        vh29
```

```
        adl 2
        vh33
        adl 2
        vh34
        fl vh28
vh35    lsr vh34
        vh36
vh36    j5 p
        p 0
vh38    vx10 p
        /p/
        cmp j5
        fl vh39
        fl vh80
        fl vh39
vh39    qq2 p
vh40    vx10 pd
        p qq2
        adl 2
        vh38
        adl 2
        vh40
jhb     fl vh38
        j9 p
        cmp j151
        fr ersp
        fr vh44
        p bc10
vh41    -j5 pd
        p bc11
        cmp j151
        fl vh42
        fl vh42
        bc10 p
        mp bc11
        p bc10
        bc11 p
        fl vh41
vh42    j3 p
        j2 pd
        cmp j151
        fr ersp
        fl vh54
vh43    p bc12
        -j5 pd
        p bc13
        cmp j151
        fl vh55
        fl vh55
        bc12 p
        mp bc13
        p bc12
        bc13 p
vh44    fr vh43
        j5 p
        p bc10
        fl vh42
vh54    j5 p
        p bc12
vh55    bc12 p
        p/bc10
        p/ qq2
        p qq3
        p vvx1
vh45    c100 p
```

```
        /p/
        cmp j5
        fl vh46
        fl vh48
        fl vh46
vh46    c100 p
        mp qq3
vh47    p vvx2
        adl 2
        vh45
        adl 2
        vh46
        adl 2
        vh47
        fl vh45
vh48    rep 6
        nr eop
vh49    0
        0
vh50    vvx1 p
vh51    pt 11
        adl 2
        vh50
        fr vh49
j2      080000
        0
        000100
        0
j3      080000
        0
        000200
        0
j4      080000
        0
        000100
        0
j150    0c0000
        0
        4 zz
        0
j5      080000
        0
        1 zz
        0
j6      080000
        0
        2 zz
        0
j7      0c0000
        0
        2 zz
        0
j8      080000
        0
        3 zz
        0
j9      0
        0
        0
        0
j10     0
        0
        0
        0
j11     0
```

```
      0
      0
      0
j12   0
      0
      0
      0
j13   0
      0
      0
      0
j14   0
      0
      0
      0
j15   0
      0
      0
      0
j16   0
      0
      0
      0
j17   0
      0
      0
      0
j18   0
      0
      0
      0
j19   0
      0
      0
      0
j20   0
      0
      0
      0
j21   0
      0
      0
      0
j22   0
      0
      0
      0
j23   0
      0
      0
      0
j24   0
      0
      0
      0
j25   0
      0
      0
      0
j26   0
      0
      0
      0
j27   0
      0
```

```
      0
      0
j28   0
      0
      0
      0
j100  0
      0
      0
      0
jq20  0
      0
      0
      0
jq30  0
      0
      0
      0
jq31  0
      0
      0
      0
j32   0
      0
      0
      0
j33   0
      0
      0
      0
j34   0
      0
      0
      0
j35   0
      0
      0
      0
j36   0
      0
      0
      0
j37   0
      0
      0
      0
j38   0
      0
      0
      0
j39   0
      0
      0
      0
j40   0
      0
      0
      0
j41   0
      0
      0
      0
j42   0
      0
      0
```

	0
j45	0
	0
	0
	0
j44	0
	0
	0
	0
	0
j43	0
	0
	0
	0
j101	0
	0
	0
	0
j102	0
	0
	0
	0
j103	0
	0
	0
	0
j104	0
	0
	0
	0
j105	0
	0
	0
	0
j106	0
	0
	0
	0
j120	0
	0
	0
	0
j121	0
	0
	0
	0
j122	0
	0
	0
	0
j123	0
	0
	0
	0
yz12	0
	0
	0
	0
j107	0
	0
	0
	0
j111	0
	0
	0

	0
j112	0
	0
	0
	0
j108	0
	0
	0
	0
wz10	0a3d70
	a3d70a
	fffa00
	0
j109	0
	0
	0
	0
j110	0
	0
	0
	0
j113	0
	0
	0
	0
j114	0
	0
	0
	0
yz10	0
	0
	0
	0
yz11	0
	0
	0
	0
yz13	0
	0
	0
	0
yz14	0
	0
	0
	0
j126	0afec0
	0
	ffdc00
	0
wy10	0
	0
	0
	0
wy11	0
	0
	0
	0
wy12	0
	0
	0
	0
wy13	0
	0
	0
	0

```
wy14    0
        0
        0
        0
cx10    0
        0
        0
        0
cq10    0
        0
        0
        0
qc10    0
        0
        0
        0
qc12    0
        0
        0
        0
qc11    0
        0
        0
        0
qq2     0
        0
        0
        0
bc13    0
        0
        0
        0
bc10    0
        0
        0
        0
bc12    0
        0
        0
        0
bc11    0
        0
        0
        0
qq3     0
        0
        0
        0
j151    0
        0
        0
        0
w18     j400 zz
        0
w23     j300 zz
        0
w9      j900 zz
        0
w91     j901 zz
        0
bgn     rd psm
        lo i400
400     t psm
        j2 8
        t prep
```

```
        vh48 1
        040083
        240030
        lo kl
psm     0
        0
        0
        0
        0
        0
        0
        0
        0
        0
        0
        0
        0
        0
        0
        0
prep    0
        0
wj71    0
        0
apne    240081
        0400fd
apnf    apnf a
        ro 512
        jq30 p
        p jq31
eop     fl k12
        stop
nrun    240083
        040030
        lo bgn
        0
fset    pc 37
        pc 37
        pc 7
        pc 4
        pc 4
        j2 p
        0300fl
        pc 37
        pc 6
        pc 4
        pc 4
        j3 p
        0300fl
        pc 37
        pc 8
        pc 4
        pc 4
        j4 p
        0303fl
        pc 37
        pc 37
        pc 37
        pc 16
        pc 16
        pc 16
        pc 16
        pc 41
        j28 p
        fl k70
```

```
vh80    j17 p
        cmp j151
        fr ersp
        fr vh83
vh81    fl vh84
        -j5 pd
        p hal1
        cmp j151
        fr vh82
        fr vh82
        hal0 p
        mp hal1
        p hal0
        hal1 p
vh82    fr vh81
        qq2 p
        hal0 pd
        p qq2
ersp    fr jhb
        nl vh83
vh83    0bad82
        j5 p
        p hal0
        fr vh82
hal0    0
        0
        0
        0
hal1    0
        0
        0
        0
vh84    j17 p
        jq30 pd
        p hal0
        fr vh81
dj50    jq30 p
        p jq31
        j150 p
        p j111
dj42    p j112
        j151 p
        p j113
        j5 p
dj41    p gg11
dj51    j9 p
        p j110
dj52    j111 p
        -j6 pd
        p j111
dj53    j110 p
        -j111 pd
dj40    fcl dj74
dj54    j112 p
        p jq31
        nl dj55
dj55    dj57 a
        lo dj56
dj56    lal k11
        il k11
dj57    dj57 a
        lo 512
dj58    fl k10
        nl dj59
dj59    dj61 a
        lo dj60
dj60    lal k9
        il k9
dj61    dj61 a
        lo 512
dj62    fl k8
        j39 p
        -j28 pd
        j113 pd
        p j114
        -j101 p
        p/j114
        p j113
dj63    p gg10
        p nlst
        adl 4098
        dj63
        j112 p
        -j6 pd
        p j112
        fl dj52
dj64    lsl wj65
        dj40
        j112 p
        j6 pd
        p j111
        jq30 p
        p j110
        fl dj52
wj65    dj65 zz
        0
dj65    adl 2
        dj63
        lsl dj63
        dj70
dj70    0 p
        cmp j5
        fl djf6
        fl dj72
djf6    p j901
        skip
        adl 2
        djf6
        adl 2
        dj70
        fl dj70
        0
dj72    lsl wj71
        dj71
        lsr w9
        dj73
dj71    adr 0
        dj73
dj73    j5 p
        p 0
        fl dj11
k79     j160 p
        j5 pd
hov7    p j160
        cmp j7
        fl k69
        fl k69
        fl kv70
j160    0
        0
```

```
      0
      0
kv70  wz10 p
      p/kten
      p wz10
      j151 p
      p j160
      flhov5
kten  080000
      0
      2 zz
      0
      0
      0
      0
      0
dj74  nlst p
      /p/
      p n'su
      lsl wj74
      dj40
      lsl wj41
      dj41
      lsl wj63
      dj63
      j150 p
      kten pd
      p wl50
      p j111
      fl dj42
wj74  dj75 zz
      0
wj41  gg11 zz
      0
wj63  gg10 zz
      0
wl50  0
      0
      0
      0
nlst  0
      0
      0
      0
nlsu  0
      0
      0
      0
dj75  -nlsu p
      /nlst/pd
      /p/
      cmp ntol
      fl dj64
      fl dj64
      nlst p
      /p/
      p nlsu
      lsl wj41
      dj41
      lsl wj63
      dj63
      wl50 p
      kten pd
      p wl50
      p j111
      fl dj42
ntol  0afec0
      0
      ffdc00
      0
ltcl  j28 p
      /p/
      p j29
      lsr mbl
      k62
      fl k79
mbl   ltc2 zz
      0
ltc2  j28 p
      /p/
      -j29 pd
      /p/
      -ltol pd
      fcl i100
      fl fset
      0
100   j28 p
      /p/
      p j29
      fl k79
ltol  0afec0
      0
      ffdc00
      0
j29   0
      0
      0
      0
kfdg  080000
      0000f0
      1 zz
      0
yadl  nrlb p
      -j5 pd
      fcl zin
      fl vav
vav   j4 p
      /p/
      cmp m992
      fl k3
      fl gnul
      fl gnul
zin   j4 p
      /p/
      cmp m992
      fl k3
      fl gnu5
      fl gnu5
gnul  j3 p
      -j2 pd
      mp j6
      j5 pd
      p m9r2
      mp m9r2
      p m9r3
      mp m9r2
      p m9r4
      mp m9r2
      p m9r5
      mp m9r2
      p m9r6
```

```
      mp m9r2
      p m9r7
      mp m9r2
      p m9r8
      j4 p
      mp j4
      p m9ra
      mp j4
      p m9rb
      mp j4
      p m9rc
      mp j4
      p m9rd
      j2 p
      mp j2
      p m9re
      mp m9re
      p mare
gnu2  j4 p
      mp m9r2
      p m9rf
      m9r3 p
      m8r0 pd
      p/ m8rl
      p m9rg
      m9re p
      mp m8r3
      p m9rh
      -m9rh p
      m9r3 pd
      m8r2 pd
      mp m9r2
      p/ m8r4
      p/ j4
      p m9rj
      m9r3 p
      mp m8r5
      m9r5 pd
      m8r6 pd
      mp m8r0
      p m9rk
      m9r3 p
      j5 pd
      mp m9re
      mp m8r8
      p m9rh
      -m9rh p
      m9rk pd
      p/ m8r7
      p/ m9ra
      p m9rn
      m9r6 p
      mp m8ra
      p m9rh
      m9r4 p
      mp m8rb
      p m9rk
      m9r2 p
      mp m8rc
      m9rk pd
      m9rh pd
      p/ m8r9
      p/ m8r9
      p m9rq
      m9r2 p
```

```
mp m8re
p m9rh
m9r4 p
mp m8rd
m9rh pd
mp m9re
p/ m8r9
p m9rt
m9r2 p
mp mare
p/ m8rl
-m9rt pd
m9rq pd
p/ m9rb
p m9ru
m9r7 p
mp m8rg
p m9rh
m9r5 p
mp m88f
p m9rk
m9r3 p
mp clal
m8rh pd
m9rk pd
m9rh pd
p/ m8rf
p/ m8rf
p m9rt
m9r5 p
mp m8rn
p m9rh
m9r3 p
mp m8rj
p m9rk
m8rp pd
m9rh pd
mp m9re
p/ m8rk
p m9rq
m9r3 p
j5 pd
mp mare
mp m991
p/ m8rl
-m9rq pd
m9rt pd
p/ m9rc
p m9rw
m9r8 p
mp m8rq
p m9rh
m9r6 p
mp m8rs
p m9rk
m9r4 p
mp m8rt
p m9rq
m9r2 p
mp m8ru
m9rq pd
m9rk pd
m9rh pd
p/ m8r7
p/ m8r7
```

```
        p m9rh
        m9r6 p
        mp m8rv
        p m9rk
        m9r4 p
        mp m8rw
        p m9rq
        m9r2 p
        mp m8ry
        m9rq pd
        m9rk pd
        mp m8re
        p/ m8r3
        p/ m8r7
        p m9rk
        m8rz p
        mp m9r4
        p m9rq
        m9r2 p
        mp m9r0
        m9rq pd
        mp mare
        p/ m8rk
        p m9rq
        m9r2 p
        mp m9re
        p/ m9rl
        p m9rt
        -m9rt p
        m9rq pd
        -m9rk pd
        m9rh pd
        p/ m9rd
        p m9rx
        fl hov 1
gnu3    m9rf p
        m9re pd
        -m9rg pd
        -m9rj pd
        -m9rn pd
        -m9ru pd
        -m9rw pd
        -m9rx pd
gnu4    p j28
        fl k3
gnu5    j4 p
        mp j4
        p g3f0
        mp j4
        p g3fl
        mp j4
        p g3f2
        mp j4
        p g3f3
        j3 p
        -j2 pd
        mp j6
        j2 pd
        j5 pd
        p g3f4
        mp g3f4
        p g3f5
        mp g3f5
        p g3f6
        mp g3f5
        p g3f7
        j2 p
        mp j2
        p g3f8
        mp g3f8
        p g3f9
        mp g3f8
        p g4f0
gnu6    g3f4 p
        mp j4
        mp j6
        p g4fl
        g3f5 p
        -g3f8 pd
        j5 pd
        p/ j6
        p g4f2
        p/ g2ck
        p/ j4
        mp g3f4
        p g4f3
        g3f6 p
        mp m8r0
        p g5a0
        g3f5 p
        mp m992
        p g5al
        g3f5 p
        mp m991
        j5 pd
        mp g3f8
        mp j6
        p g5a2
        -g5a2 p
        g3f9 pd
        j5 pd
        g5al pd
        g5a0 pd
        p/ m8r4
        p/ g3f0
        p g4f4
        g3f6 p
        mp m8ra
        p g5a0
        g3f5 p
        mp g2c2
        p g5al
        g3f5 p
        mp g2cl
        g2c4 pd
        mp g3f8
        mp j6
        p g5a2
        g3f9 p
        mp g2c5
        -g5a2 pd
        m8rd pd
        g5al pd
        g5a0 pd
        mp g3f4
        p/ m8rk
        p/ g3fl
        p g4f5
        g3f7 p
        mp m8rg
```

```
p g5a0
g3f6 p
mp g2c3
p g5a1
g3f5 p
mp g2c6
p g5a2
g3f6 p
mp m992
p g5a3
g3f5 p
mp g2c1
m991 pd
g5a3 pd
mp g3f8
mp m992
p g5a3
g3f5 p
mp g2c5
g2c8 pd
mp g3f9
mp m991
p g5a4
g4f0 p
mp j6
p g5a5
-g5a5 p
g5a4 pd
-g5a3 pd
g2c7 pd
g5a2 pd
g5a1 pd
g5a0 pd
p/ m8r7
p/ g3f2
p g4f6
g3f7 p
mp m8rq
p g5a0
g3f6 p
mp g2c9
p g5a1
g3f5 p
mp g2ca
p g5a2
g3f6 p
mp g2cc
p g5a3
g3f5 p
mp g2cd
g2ce pd
g5a3 pd
mp g3f8
p g5a3
g3f5 p
mp g2cf
g2cg pd
mp g3f9
p g5a4
g4f0 p
mp g2ch
p g5a5
-g5a5 p
g5a4 pd
-g5a3 pd
```

```
       g2cb pd
       g5a2 pd
       g5a1 pd
       g5a0 pd
       mp g3f4
       p/ g2cj
       p/ g3f3
       fl hov3
gnu7   -g3f0 p
       g4f1 pd
       -g4f2 pd
       -g4f3 pd
       -g4f4 pd
       -g4f5 pd
       -g4f6 pd
       -g4f7 pd
gnu8   p j28
       fl k3
m8r0   0a0000
       0
       3 zz
       0
m8r1   080000
       0
       4 zz
       0
m8r2   0b0000
       0
       4 zz
       0
m8r3   080000
       0
       6 zz
       0
m8r4   080000
       0
       7 zz
       0
m8r5   0d0000
       0
       5 zz
       0
m8r6   0a8000
       0
       5 zz
       0
m8r7   080000
       0
       11 zz
       0
m8r8   0c0000
       0
       9 zz
       0
m8r9   080000
       0
       8 zz
       0
m8ra   084000
       0
       6 zz
       0
m8rb   0c7400
       0
       11 zz
```

```
      0
m8rc  0afa80
      0
      13 zz
      0
m8rd  094000
      0
      6 zz
      0
m8re  0a7000
      0
      8 zz
      0
m8rf  080000
      0
      9 zz
      0
m8rg  0fc000
      0
      6 zz
      0
nrlb  0 zz
      0
      0
      0
m8rt  0fedf9
      0
      20 zz
      0
m8ru  088d0f
      c00000
      22 zz
      0
m8rv  0b3580
      0
      13 zz
      0
m8rw  0f91f0
      0
      17 zz
      0
m8ry  091f8e
      0
      19 zz
      0
m8rz  0b1800
      0
      9 zz
      0
m9r0  0bc200
      0
      15 zz
      0
m9r1  080000
      0
      5 zz
      0
m991  0c0000
      0
      2 zz
      0
m9r2  0 zz
      0
      0
      0
```

```
m9r3  0 zz
      0
      0
      0
m88f  09a600
      0
      13 zz
      0
clal  0a93f0
      0
      16 zz
      0
m8rh  0af8c0
      0
      15 zz
      0
m8rj  0a3c00
      0
      11 zz
      0
m8rk  080000
      0
      10 zz
      0
m8rn  0e6000
      0
      7 zz
      0
m8rp  0b7c00
      0
      10 zz
      0
m8rq  083c00
      0
      10 zz
      0
m8rs  0f0590
      0
      16 zz
      0
m992  0a0000
      0
      4 zz
      0
m9r4  0 zz
      0
      0
      0
m9r5  0 zz
      0
      0
      0
m9r6  0 zz
      0
      0
      0
m9r7  0 zz
      0
      0
      0
m9r8  0 zz
      0
      0
      0
      0
```

```
m9r9    0 zz
        0
        0
        0
m9ra    0 zz
        0
        0
        0
m9rb    0 zz
        0
        0
        0
m9rc    0 zz
        0
        0
        0
m9rd    0 zz
        0
        0
        0
m9re    0 zz
        0
        0
        0
m9rf    0 zz
        0
        0
        0
m9rg    0 zz
        0
        0
        0
m9rh    0 zz
        0
        0
        0
m9rj    0 zz
        0
        0
        0
m9rk    0 zz
        0
        0
        0
m9rn    0 zz
        0
        0
        0
m9rq    0 zz
        0
        0
        0
m9rt    0 zz
        0
        0
        0
mare    0 zz
        0
        0
        0
m9ru    0 zz
        0
        0
        0
m9rw    0 zz
```

```
        0
        0
        0
m9rx    0 zz
        0
        0
        0
        0
g2c1    0b8000
        0
        5 zz
        0
g2c2    0e4000
        0
        7 zz
        0
g2c3    0aa000
        0
        9 zz
        0
g2c4    0c8000
        0
        5 zz
        0
g2c5    0d0000
        0
        4 zz
        0
g2c6    0ef000
        0
        8 zz
        0
g2c7    0e0000
        0
        4 zz
        0
g2c8    0c0000
        0
        3 zz
        0
g2c9    081580
        0
        13 zz
        0
g2ca    0a3280
        0
        17 zz
        0
g2cb    0fc400
        0
        10 zz
        0
g2cc    0eac00
        0
        10 zz
        0
g2cd    0ea600
        0
        12 zz
        0
g2ce    0c6e00
        0
        11 zz
        0
g2cf    0e8800
```

```
        0
        9 zz
        0
g2cg    09ec00
        0
        10 zz
        0
g2ch    0d4000
        0
        6 zz
        0
g2cj    080000
        0
        14 zz
        0
g2ck    080000
        0
        3 zz
        0
g3f0    0 zz
        0
        0
        0
g3f1    0 zz
        0
        0
        0
g3f2    0 zz
        0
        0
        0
g3f3    0 zz
        0
        0
        0
g3f4    0 zz
        0
        0
        0
g3f5    0 zz
        0
        0
        0
g3f6    0 zz
        0
        0
        0
g3f7    0 zz
        0
        0
        0
g3f8    0 zz
        0
        0
        0
g3f9    0 zz
        0
        0
        0
g4f0    0 zz
        0
        0
        0
g4f1    0 zz
        0
        0
        0
g4f2    0 zz
        0
        0
        0
g4f3    0 zz
        0
        0
        0
g4f4    0 zz
        0
        0
        0
g4f5    0 zz
        0
        0
        0
g4f6    0 zz
        0
        0
        0
g4f7    0 zz
        0
        0
        0
g5a0    0 zz
        0
        0
        0
g5a1    0 zz
        0
        0
        0
g5a2    0 zz
        0
        0
        0
g5a3    0 zz
        0
        0
        0
g5a4    0 zz
        0
        0
        0
g5a5    0 zz
        0
        0
        0
m9rf    p
|p|
p       g3f0
m9rx    p
|p|
```

```
             p         g3f1
             g3f0      p
             mp        hov2
             -g3f1     pd
             fcl       gnu3
             fl        k3
hov2         0a3d70
             a3d70a
             fff900
             000000
hov3         p         g4f7
             g3f0      p
             |p|
             p         m9rf
             g4f7      p
             |p|
             p         m9rg
             m9rf      p
             mp        hov2
             -m9rg     pd
             fcl       gnu7
             fl        k3
hov4         pc        37
             pc        16
             pc        16
             pc        16
             pc        41
             j126      p
             pt        11
             nl        k68
hov5         j126      p
             mp        m8ro
             mp        j6
             p         j126
             p         ℓtol
             lsr       hov6
             hov7
             fl        k69
```

NOTE: Reserve in the memory 200 consecutive floating point number locations for each of the starting addresses listed below. There will then be in the memory 2200 floating point number reservations. For example, beginning with address b100 there will follow b101, b102, etc., each of which is reserved for storage.

b100

c100

j300

j400

j900

k300

vx10

vvx1

v100

w100

y100

* * *

APPENDIX C

REPRODUCTION OF PROGRAM IN MACHINE LANGUAGE FORM

The following list of commands is a Narec "machine language" translation of the symbolic program appearing in Appendix B. It is to be noted that list contains both fixed-point and floating-point commands. Two columns of numbers appear in the list. The left hand column is extraneous to the actual computation schedule. It is used only for visual identification of the lines in the right hand column which constitutes the true read-in. The left-hand column is not read into the computer. However the lines in the program must be read in consecutively beginning at address 0400.

Each line of command consists of twelve hexadecimal digits. They are interpreted by the computer logic from left to right as follows:

digits 1 through 4 represent one address in memory

digits 5 and 6 represent a fixed-point or floating point order

digits 7 through 10 represent one address in memory

digits 11 and 12 represent a fixed-point or gloating-point order.

Note that the program starts in fixed-point. To enter floating-point as a particular address one uses the standard form

(address) 50 0200 11

To exit floating point into fixed point one uses the symbols E4 or E5.

The symbol || indicates the division of a Narec word between the left order and the right order.

The symbol * means "skip this order."

APPENDIX D

TYPICAL OUTPUT SHEET

A typical output sheet is reproduced below with added description of the meaning of the numbers. A number without a decimal point is interpreted as fraction followed by an exponent. For example

+65511109498   +02

is

+65.511109 498

nnnn

0000

m +0.000000000 -2
ℓ +6.
h +6.600

0000	+65511109498	+02	Eigenvalue
0001			
0000	+10000000827	-07	Precision of Eigenvalue
0002	+10936175364	-01	Prolate Spheroidal Wave Expansion Constants
0003	+96013725118	-01	
0004	+38634237703	+00	
0005	+83439671453	+00	
0006	-39362130016	+00	
0007	+73217232273	-01	
0008	-78111671012	-02	
0009	+55342208234	-03	
0010	-28240348044	-04	
0011	+10937705008	-05	
0012	-33366809971	-07	
0013	+82430513625	-09	
0014	-16852080723	-10	
0015	+29012288326	-12	
0016	-42668929896	-14	
0017	+54260160987	-16	
0018	-60277682314	-18	
0019	+59019532925	-20	
0020	-51329455278	-22	
0021	+39924268608	-24	
0022	-27941066789	-26	
0023	+17690620699	-28	
0024	-10182634883	-30	
0025	+53519977323	-33	
0026	-25790419362	-35	
0027	+11436285148	-37	
0028	-46823211649	-40	
0029	+17755655572	-42	
0030	-62540105973	-45	
0031	+20515535018	-47	
0032	-62832443777	-50	
0033	+18007915429	-52	
0034	-48401306663	-55	
0035	+12224784786	-57	
0036	-29069549227	-60	
0037	+65196116625	-63	
0038	-13814023500	-65	
0039	+27696278773	-68	
0040	-52622999510	-71	
0041	+94885173204	-74	
0042	-16258237917	-76	
0043	+26506679241	-79	
0044	-41169066792	-82	
0045	+60984915784	-85	
0046	-86255720730	-88	
0047	+11660682339	-90	
0048	-15082269790	-93	
0049	+18682422465	-96	
0050	-22183121071	-99	

APPENDIX E
NOTES

The computation results shown in Appendix D requires 116 seconds for completion. This time depends largely on the precision of the eigenvalue deemed acceptable. The precision is controlled in turn by the number of times the corrections δA_{ml} is applied to the eigenvalue A_{ml}. In this program the number of corrections is set at four. If greater precision is demanded, one enters the program on the right side of location 069C and replaces the existing address (=058F) containing the existing number of corrections (minue one) with another address containing a larger number. For example if it is desired to apply 9 corrections to A_{ml} instead of 4 corrections one scans through the program to find the fixed number "8." This is seen to be at address 0793. Hence location 069C is changed to read

069F F8 0793 DC

The computation of A_{ml} is highly sensitive to the starting value inserted in address 05B9. When h is large (i. e.>10) we use a formula by Meixner (5) to help fix a good approximation for A_{ml}. The Meixner approximation has eight terms (prolate) or seven terms (oblate). In the program the first term in either case is multiplied by a small number ($10^{-2}/2$) fixed in location 086B and 086C. The last term is then compared with the resultant multiplication. If the last term is 1/100 or less of the first term then the Meixner approximation is used. Otherwise the standard approximation (i.e. approximation $h \to 0$) is used.

0400	063810000000	Go to left order of 0638 || *
0401	040150020011	Enter floating point
0402	0585e60583ea	
0403	0593e80585e6	
0404	058be90595e8	
0405	0585e6058deb	
0406	0597e8058bea	
0407	0599e80591e9	
0408	059be80583e6	
0409	0583eb058deb	
040a	059de8058deb	
040b	059fe80583e6	
040c	058be905a1e8	
040d	0583e6058deb	
040e	05a3e8058bea	
040f	05a5e8058de9	Compute "Starting" $A_{m\ell}$ =
0410	05a7e80591ea	$\ell(\ell+1) + h^2\left[\frac{2\ell(\ell+1) - 2m^2 - 1}{(2\ell-1)\ (2\lambda+3)}\right] + 1.0000....050$
0411	05a9e80591e9	
0412	058de905abe8	
0413	0587e60587eb	
0414	05ade8058dec	
0415	05afe805ade6	
0416	05adeb05b1e8	
0417	0585e60595eb	
0418	05b3e80597e6	
0419	0595eb059dea	
041a	058bea05b5e8	
041b	0599e6059beb	
041c	05b7e805b5e6	
041d	05b7ec05adeb	
041e	05b3e906d4e9	
041f	05b9e806d6e2	Store $A_{m\ell}$ in 05B9 || Go to left order 06D6
0420	0593e605bbe8	Is ℓ - m an odd number or an even number?
0421	05bbe6058bdc	

0422	0426c20428c2	
0423	0424e20000e0	
0424	05bbe6058dea	
0425	05bbe80421e2	
0426	05bde605bfe8	(ℓ - m is even) bring up zero ‖ Store zero in 05BF
0427	05c1e80445e2	Store zero in 05C1 ‖ Go to 0445
0428	058be605bfe8	(ℓ - m is odd) Bring up "1" ‖ Store "1" in 05BF
0429	05c1e80445e2	Store "1" in 05C1 ‖ Go to 0445
042a	0583e605c1e9	
042b	05c3e8058be9	
042c	05c3eb05c5e8	
042d	059fe6058bea	
042e	05c7e805c1e6	
042f	058deb05c9e8	
0430	05a5e605c9e9	Compute $\tau_n^m = (m+n)(m+n+1) + \frac{h^2}{2}\left[1 - \frac{4m^2-1}{(2m+2n-1)(2m+2n+3)}\right]$
0431	05cbe80591e9	
0432	05cbeb05cde8	
0433	05c7e605cdec	
0434	05cfe8058be6	
0435	05cfea05afeb	
0436	05c5e905d1e8	
0437	0000e30000e0	
0438	05c1e6058bea	
0439	05d3e805a3e6	
043a	05c1e905d5e8	
043b	058bea05d7e8	
043c	05c1e905d9e8	
043d	058de905dbe8	
043e	0591ea05dee8	
043f	05c1e605d3eb	
0440	05d5eb05d7eb	
0441	05d9ec05d9ec	
0442	05deec05dbec	
0443	05b1eb05e0e8	
0444	0000e30000e0	

0445	058bc61900e8	
0446	0593e60591e9	
0447	05e2e805bfe6	
0448	05e4e805c1e8	
0449	05c4e6058de9	
044a	05e4e805e2dc	
044b	044de20456e2	
044c	0456e20000e0	
044d	044ee40000e0	
044e	045050044f10	
044f	044420044422	
0450	045050020010	
0451	0438e205e0e6	
0452	0890c80000e0	
0453	0002d0045200	
0454	05c1e6058de9	
0455	05c1e80449e2	
0456	05bfe605c1e8	
0457	05e6e80000e0	Compute $\left[N_n^m+2\right]_2 = \dfrac{-\beta_n^m}{N_n^m} + A_m\ell - \tau_n^m$
0458	05e6e6058de9	
0459	05e6e805e2dc	
045a	045ce20465e2	
045b	0465e20000e0	
045c	045de40000e0	
045d	045f50045e10	
045e	043720043722	
045f	045f50020010	
0460	042ae205d1e6	
0461	0a20e80000e0	
0462	0002d0046100	
0463	05c1e6058de9	
0464	05c1e80458e2	
0465	05bfe605c1e8	
0466	05e8e80000e0	
0467	05e8e6058de9	
0468	05e8e805e2dc	Finished?
0469	046be20474e2	No: Go to 046B || Yes: Go to 0474
046a	0474e20000e0	Yes: Go to 0474 || *
046b	0890e70000e0	
046c	1900ec0000e0	
046d	0a20ea05b9e9	
046e	1902e805eae8	|| Store $\left[N_{\ell-m+2}^m\right]_2$ in 05EA
046f	0002d0046b00	
0470	0002d0046c00	
0471	0002d0046d00	
0472	0002d0046e00	
0473	087FE20000E0	

0474	0593e6058de9		
0475	05ece805f2e8	Compute $\dfrac{\beta^m_{\ell-m-2x}}{(N^m_{\ell-m-2x})^2}$ for $x = 1, 2, \ldots$	
0476	058dea049fe8		
0477	05bfe605cee8		
0478	05eee6058de9		
0479	05eee805f2dc		‖ Finished?
047a	047ce20485e2	No: Go to 047C	‖ Yes: Go to 0485
047b	0485e20000e0	Yes: Go to 0485	
047c	0892e60000e0		
047d	1902ec0000e0		
047e	1902ec0000e0		
047f	0bb0e80000e0		
0480	0002d0047c00		
0481	0002d0047d00		
0482	0002d0047f00		
0483	0002d0047e00		
0484	0478e20000e0		
0485	047fd8048700		
0486	058be60000e0		
0487	0000e80000e0		
0488	0bb0e60000e0		
0489	058bdc048be2	Finished?	‖ No: Go to 048B — Outer Loop
048a	04a1e2048be2	Yes: Go to 04A1	‖ No: Go to 048B
048b	0488d8049100		
048c	0488d8048e00		
048d	0002d0049100		
048e	0000e60d40e8		
048f	05bfe605f0e8	Compute $\left[\dfrac{\beta^m_{\ell-m}}{(N^m_{\ell-m})^2}\right]\left[\dfrac{\beta^m_{\ell-m-2x}}{(N^m_{\ell-m-2x})^2}\right]$ etc.	
0490	0d40e60000e0		
0491	0000eb0000e0		
0492	0d40e80000e0		
0493	05f0e6058de9		
0494	05f0e8049fdc	‖ Finished?	Inner Loop
0495	0497e20499e2	No: Go to 0497	‖ Yes: Go to 0499
0496	0499e20000e0	Yes: Go to 0499	‖ *
0497	0002d0049100		
0498	0490e20000e0		
0499	0002d0048800		
049a	0002d0049200		
049b	0002d1048e00		
049c	0002d0049000		
049d	049fe6058dea		
049e	049fe80488e2		
049f	000000000000		
04a0	000000000000		

04a1	0492d904a200	
04a2	058bc60000e8	
04a3	0d40e6058bdc	
04a4	04a6e204abe2	Compute $\left[1 + \frac{\beta^m_{\ell-m}}{(N^m_{\ell-m})^2} + \frac{\beta^m_{\ell-m}\,\beta^m_{\ell-m-2}}{(N^m_{\ell-m})^2\,(N^m_{\ell-m-2})^2} + \cdots\right]$
04a5	04a6e20000e0	
04a6	05f4e60000e0	
04a7	0d40e905f4e8	Store computation in 05F4*
04a8	0002d004a300	
04a9	0002d004a700	
04aa	04a3c20000e0	
04ab	05f4e6058be9	
04ac	05f4e80000e0	
04ad	0593e605f6e8	
04ae	05f6e6058de9	
04af	05f6e805f8e8	Store max. (n + 2x) in 05F8
04b0	05fae805c1e8	
04b1	04b2e40000e0	
04b2	04b45004b310	
04b3	044420044422	
04b4	04b450020010	
04b5	0438e204b6e4	
04b6	04b85004b710	
04b7	043720043722	Compute $\left[N^m_{n+2x}\right]_1$ for determining (n + 2x) max.
04b8	04b850020010	
04b9	042ae205d1e6	
04ba	05b9ea05fce8	
04bb	05fceb0000f9	
04bc	0600e805e0e7	
04bd	05fcec0000f9	
04be	0600dc04c0e2	Finished? Yes: Go to 04C0
04bf	04c0e204aee2	Yes: Go to 04C0 || No: Go to 04AE

04c0	05bfe605c1e8	
04c1	0593e6058dea	
04c2	0602c80000e0	
04c3	05f8e6058dea	
04c4	05f8e80000e0	
04c5	0602e605f8ea	
04c6	04dbf40000e0	If finished go to 04DB || *
04c7	05fae605c1e8	
04c8	04c9e40000e0	
04c9	04cb5004ca10	
04ca	044420044422	
04cb	04cb50020010	
04cc	0438e204cde4	
04cd	04cf5004ce10	Compute $\frac{\left[N^m_{\ell-m+2x}\right]_1^2}{\beta^m_{\ell+m+2x}}$ for $x = 1, 2, \ldots$
04ce	043720043722	
04cf	04cf50020010	
04d0	042ae205d1e6	
04d1	05b9ea0604e9	
04d2	0606e805e0e7	
04d3	0606ec0604e8	|| Store $\left[N^m_{\ell-m+2}\right]_1$ in 0604
04d4	0604e60604eb	
04d5	05e0ec0000e0	
04d6	0ed0e80000e0	Store $\left(N^m_{\ell-m+2x}\right)^2 / \beta^m_{\ell-m+2x}$ in OEDO || *.
04d7	05fae6058dea	
04d8	05fae80000e0	
04d9	0002d004d600	
04da	04c3e20000e0	

04db	04d6d904dc00	
04dc	058be60000e8	
04dd	058be60608e8	
04de	0ed0e6058bdc	
04df	04e1e204efe2	
04e0	04e1e20000e0	
04e1	0ed0e6058bdc	
04e2	04e1e204eae2	
04e3	04e1e20000e0	
04e4	0608e60000e0	
04e5	0ed0eb0000e0	
04e6	1060e80008e8	Compute $\frac{(N^m_{\ell-m+2})^2}{\beta^m_{\ell-m+2}} + \frac{(N^m_{\ell-m+2})^2}{\beta_{\ell-m+2}} \cdot \frac{(N^m_{\ell-m+4})}{\beta_{\ell-m+4}} + \ldots$
04e7	0002d004e100	
04e8	0002d004e500	
04e9	04e1e20000e0	
04ea	0002d004dc00	
04eb	04ded804e100	
04ec	04ded804e500	
04ed	0002d004e600	
04ee	04dde20000e0	
04ef	04e6d904f000	
04f0	058be60000e8	
04f1	1060e6058bdc	|| Finished?
04f2	04f4e204f9e2	No: Go to 04F4 || Yes: Go to 04F9
04f3	04f4e20000e0	No: Go to 04F4
04f4	060ae60000e0	|| Store $\Sigma\ [(N^m_{\ell-m+2})^2/\beta_{\ell-m+2}]\ [(N^m_{\ell-m+4})^2/\beta^m_{\ell-m+4}]$
04f5	1060e9060ae8	etc. in 060A*
04f6	0002d004f100	
04f7	0002d004f500	
04f8	04f1e20000e0	

04f9	05f4e6060ae9	Bring up $1 + \Sigma\, \beta^m_{\ell-m}/(N^m_{\ell-m})^2$ etc ‖ Add $\Sigma\, (N^m_{\ell-m+2})^2/\beta^m_{\ell-m+2}$
04fa	060ce80604e6	Store in 060C ‖ Bring up $[N^m_{\ell-m+2}]_1$
04fb	05eaea060cec	Subtract $[N^m_{\ell-m+2}]_2$ ‖ Divide by 060C to form $\delta A_{m\ell}$
04fc	060ee805b9e9	Store $\delta A_{m\ell}$ in 060E ‖ Add $A_{m\ell}$
04fd	05b9e8060ee6	Store $\delta A_{m\ell} + A_{m\ell}$ in 05B9 ‖ Bring up $\delta A_{m\ell}$
04fe	000019 0610ea	Form abs value ‖ Subtract tolerance in 0610
04ff	069bf406c5e2	Does $A_{m\ell}$ meet precision? No: Go to 0693 ‖ Yes: Go to 06C5
0500	000bf00873e2	Output: Print to 11 significant digits ‖ Go to 0873
0501	050111050910	Go to right side of this line ‖ Go to left order of 0509
0502	0632e605e4e8	Bring up zero from 0632 ‖ Store zero in 05E4
0503	05e6e805e8e8	~ in 05E6 ‖ ~ in 05E8
0504	05eee805f0e8	~ in 05EE ‖ ~ in 05F0
0505	05f2e8060ae8	~ in 05F2 ‖ ~ in 060A
0506	05f4e805f6e8	~ in 05F4 ‖ ~ in 05F6
0507	05f8e805fae8	~ in 05F8 ‖ ~ in 05FA
0508	0604e80647e4	~ in 0604 ‖ ~ Enter fixed point at 0647
0509	050950020011	Enter floating point
050a	058be61900e8	Bring up "1" ‖ Store in 1900
050b	066ee60000e0	Dummy command
050c	050bd8046a00	Change exit from 046A to 066E
050d	0593e60591e9	Bring up n − m ‖ Add "4"
050e	05e2e80632e6	Store n−m+4 in 05E2 ‖ Bring up zero
050f	05e4e805e6e8	Store zero in 0524 ‖ Store zero in 05E6
0510	05e8e80000e0	Store zero in 05E8 ‖ *
0511	0634d8045200	
0512	0635d8046100	
0513	0634d8046b00	Compute all $[N^m_{\ell-m}]$
0514	0636d8046c00	
0515	0635d8046d00	
0516	0637d8046e00	
0517	0447e30000e0	

0518	0692d0051900	Transfer 19xx [$=N_{xxx}$] to right of 0519
0519	058be60000e8	Bring up "1" || Store "1" in 1902
051a	05bfe60612e8	Bring up zero (even) or 1 (odd) || Store in 0612
051b	0612e6058de9	Bring up zero or "1" || Add 2
051c	0612e80000e0	Store 0 (or 1) + 2x in 0612 (= counter) || *
051d	1902e60000f9	Bring up 1902 + 2x || Form abs value
051e	058bde0520e2	Finished? || No: Go to 0520
051f	052fe20520e2	Yes: Go to 052F || No: Go to 0520
0520	0612e6058deb	Bring up 0 (or 1) + 2x || Mult. by 2
0521	05a3e9058be9	Add 2m || add 1
0522	0614e8058dea	Store in 0614 || subtract 2
0523	0614eb0616e8	Multiply by 0614 || Store in 0616
0524	0612e605a3e9	Bring up 0 (or 1) 12 x || add 2m
0525	0618e8058bea	Store in 0618 || Subtract 1
0526	0618eb061ae8	Multiply by 0618 || Store in 061A
0527	0616e6061aec	Bring up 0616 || Divide by 061A
0528	05adec0000e0	
0529	1902eb0000e0	
052a	11f0e80000e0	
052b	0002d0051d00	
052c	0002d0052900	Compute all $\left[\frac{d_n}{d_{n-2}}\right]$
052d	0002d0052a00	
052e	051be20000e0	

```
052f   052ad9053000
0530   058be60000e8
0531   061ce6058be9
0532   061ce80000e0      Compute d2/do, ds/do, d6/do etc.
0533   11f0e60000f9
0534   058bdc0536e2      Finished? || No: Go to 0536
0535   053de20536e2      Yes: Go to 053D || No: Goto 0536
0536   061ce60000e0
0537   11f0eb0000e0
0538   1380e8061ce8
0539   0002d0053300
053a   0002d0053700
053b   0002d0053800
053c   0533e20000e0

053d   0538d0053e00
053e   058be60000e8
053f   05bfe6061ee8
0540   061ee6058de9
0541   061ee8061ee6      Compute (n + 2m)!/n! dn/do
0542   05a3e90620e8
0543   0622e80000e0
0544   1380e60000f9
0545   058bdc0547e2
0546   0555c20547e2      Finished? Go to 0555

0547   061ce60000f9
0548   0620dc054ae2
0549   054ee2054ee2
054a   0620e6058bea
054b   0624e80622eb
054c   0622e80624e6
054d   0620e80547e2
054e   0622e6061eec
054f   1380eb0000e0
0550   1510e80000e0
0551   0002d0054400
0552   0002d0054f00
0553   0002d0055000
0554   0540e20000e0
```

0555	0550d9055600	
0556	058be60000e8	Compute $\sum_{n} \frac{(n+2m)!}{n!} \frac{d_n}{d_o}$
0557	1510e60000f9	
0558	058bdc055ae2	Finished? || No: Go to 055A
0559	065ce2055ae2	Yes: Go to 065C || No: Go to 055A
055a	0626e60000e0	
055b	1510e90626e8	
055c	0002d0055700	
055d	0002d0055b00	
055e	0557e20593e6	Return to 0557 || Bring up "$\ell-m$"
055f	0632dc0665e3	Compare with zero || $(\ell-m) < 0$. Error: Go to right of 0665
0560	056ee3062ae8	$(\ell-m) = 0$. Go to right of 056E || $(\ell-m) > 0$. Store $(\ell-m)$ in 062A
0561	058bea062ee8	Subtract "1" || Store in 062E
0562	0632dc0566e2	Compare with zero || $(\ell-m-1) < 0$. Go to left side of 0566
0563	0566e2062ae6	$(\ell-m-1) = 0$. Go to left side of 0566 || $(\ell-m-1) > 0$. Bring up 062A
0564	062eeb062ae8	Multiply by 062E || Store in 062A
0565	062ee60561e2	Bring up 062E || Go to left side of 0561
0566	0585e60583e9	Bring up "ℓ" || add $A_{m\ell}$
0567	0632dc0665e3	Compare with zero || $(\ell-m) < 0$. Error: Go to right 0665
0568	0570e20000e0	Compute d_0 or d_1 $(\ell+m) = 0$. Go to left of 0570 || *
0569	062ce8058bea	$(\ell+m) > 0$. Store $(\ell+m)$ in 062C || Subtract "1"
056a	0628e80632dc	Store $(\ell+m-1)$ in 0628 || Compare with zero
056b	0571e20571e2	$(\ell+m-1) < 0$. Go to left of 0571 || $(\ell+m-1) = 0$. Go to left of 0571
056c	062ce60628eb	$(\ell+m-1) > 0$. Bring up 062C || Multiply by 0628
056d	062ce80628e6	Store result in 062C || Bring up 0628
056e	0569e3058be6	Go to right side of 0569 || Bring up 1
056f	062ae80566e2	Store in 062A || Go to left of 0566
0570	058be6062ce8	Bring up "1" || Store in 062C
0571	062ce6062aec	Bring up 062C || Divide by 062A
0572	0626ec0630e8	Divide by 0626 || Store $d_{0,1}$ in 0630
0573	16a0e80000e0	Store $d_{0,1}$ in 16A0 || *

0574	1380e60000f9	
0575	058bdc0577e2	Compute d_n, d_{n+2}, d_{n+4}, etc (n = 0,1 to start)
0576	057de20577e2	When finished go to 057D
0577	1380e60630eb	
0578	16a2e80000e0	
0579	0002d0057400	
057a	0002d0057700	
057b	0002d0057800	
057c	0574e20000e0	
057d	0006de064ae5	|| Finished? Yes: Go to right of 064A
057e	000000000000	
057f	16a0e60000e0	
0580	0000e00000e0	Print out d_n, d_{n+2}, d_{n+4} etc.
0581	0002d0057f00	
0582	057ee30000e0	
0583	080000000000	m
0584	000100000000	
0585	080000000000	ℓ
0586	000200000000	
0587	080000000000	h
0588	000100000000	
0589	0c0000000000	xxx in d_{xxx}
058a	000400000000	
058b	080000000000	
058c	000100000000	
058d	080000000000	"2"
058e	000200000000	
058f	0c0000000000	
0590	000200000000	
0591	080000000000	
0592	000300000000	

05b3	000000000000
05b4	000000000000
05b5	000000000000
05b6	000000000000
05b7	000000000000
05b8	000000000000
05b9	000000000000
05ba	000000000000
05bb	000000000000
05bc	000000000000
05bd	000000000000
05be	000000000000
05bf	000000000000
05c0	000000000000
05c1	000000000000
05c2	000000000000
05c3	000000000000
05c4	000000000000
05c5	000000000000
05c6	000000000000
05c7	000000000000
05c8	000000000000
05c9	000000000000
05ca	000000000000
05cb	000000000000
05cc	000000000000
05cd	000000000000
05ce	000000000000
05cf	000000000000
05d0	000000000000
05d1	000000000000
05d2	000000000000

05d3	000000000000
05d4	000000000000
05d5	000000000000
05d6	000000000000
05d7	000000000000
05d8	000000000000
05d9	000000000000
05da	000000000000
05db	000000000000
05dc	000000000000
05dd	000000005de10
05de	000000000000
05df	000000000000
05e0	000000000000
05e1	000000000000
05e2	000000000000
05e3	000000000000
05e4	000000000000
05e5	000000000000
05e6	000000000000
05e7	000000000000
05e8	000000000000
05e9	000000000000
05ea	000000000000
05eb	000000000000
05ec	000000000000
05ed	000000000000
05ee	000000000000
05ef	000000000000
05f0	000000000000
05f1	000000000000
05f2	000000000000

05f3	000000000000	
05f4	000000000000	
05f5	000000000000	
05f6	000000000000	
05f7	000000000000	
05f8	000000000000	
05f9	000000000000	
05fa	000000000000	
05fb	000000000000	
05fc	000000000000	
05fd	000000000000	
05fe	0a3d70a3d7a0	10^{-2}
05ff	fffa00000000	
0600	000000000000	
0601	000000000000	
0602	000000000000	
0603	000000000000	
0604	000000000000	
0605	000000000000	
0606	000000000000	
0607	000000000000	
0608	000000000000	
0609	000000000000	
060a	000000000000	
060b	000000000000	
060c	000000000000	
060d	000000000000	
060e	000000000000	
060f	000000000000	
0610	0afec0000000	10^{-11}
0611	ffdc00000000	
0612	000000000000	

0613	000000000000
0614	000000000000
0615	000000000000
0616	000000000000
0617	000000000000
0618	000000000000
0619	000000000000
061a	000000000000
061b	000000000000
061c	000000000000
061d	000000000000
061e	000000000000
061f	000000000000
0620	000000000000
0621	000000000000
0622	000000000000
0623	000000000000
0624	000000000000
0625	000000000000
0626	000000000000
0627	000000000000
0628	000000000000
0629	000000000000
062a	000000000000
062b	000000000000
062c	000000000000
062d	000000000000
062e	000000000000
062f	000000000000
0630	000000000000
0631	000000000000

0632	000000000000	
0633	000000000000	
0634	089000000000	
0635	0a2000000000	
0636	190000000000	
0637	190200000000	
0638	063d32063910	Read m, ℓ, h, etc. into 063D etc. || Go to 0639
0639	063d81058308	Transfer 8 data lines from 063D etc into 0583 etc.
063a	064581057d01	Transfer 1 data line from 0645 into 057D
063b	040083240030	Transfer 768 commands from 0400 etc. to 2400 etc.
063c	040110063d10	Go to left order 0401 || *
063d	000000000000	
063e	000000000000	
063f	000000000000	
0640	000000000000	
0641	000000000000	
0642	000000000000	
0643	000000000000	
0644	000000000000	
0645	000000000000	
0646	000000000000	
0647	2400810400fd	Transfer fresh program (256 lines) from 2400 to 400
0648	064850020011	Enter floating point
0649	05bfe605c1e8	Bring up "odd-even" (= "one" or "zero") || Store in 05C1
064a	0445e2000082	Go to 0445 || COMPUTATION FINISHED: STOP
064b	240083040030	
064c	063810000000	
064d	0025f30025f3	Output: Carriage return (C.R.) || C.R.
064e	0007f30004f3	Print character "m" || Space (on typewriter)
064f	0004f30583e6	Space || Bring up "m"

0650	0300f10025f3	Print value of m || C.R.
0651	0006f30004f3	Print character "ℓ" || Space
0652	0004f30585e6	Space || Bring up "ℓ"
0653	0300f10025f3	Print value of "ℓ" || C.R.
0654	0008f30004f3	Print character "h" || Space
0655	0004f30587e6	Space || Bring up "h"
0656	0303f10025f3	Print value of "h" || C.R.
0657	0025f30025f3	C.R. || C.R.
0658	0010f30010f3	Print character "0" || "0"
0659	0010f30010f3	"0" || "0"
065a	0029f305b9e6	Tab (on typewriter) || Bring up $A_{m\ell}$
065b	0500e20000e0	Go to 0500 || *
065c	05a3e60632dc	Bring up 2m || Compare 2m with zero
065d	0665e30666e3	2m is < zero. This is an error. Go to right 0665 || 2m = 0, Go to 0666
065e	066ce2058bea	2m > 0. Go to left of 066C || Subtract "1"
065f	066ae80632dc	Store 066A || Compare with zero
0660	0663e30663e3	Result is less than zero. Go to right of 0663 || Result equals zero, go to right of 0663
0661	0668e6066aeb	Result > 0. Bring up 0668 || Multiply by 066A
0662	0668e8066ae6	Store in 0668 || Bring up 066A
0663	065ee30626e6	Go to right of 065E || Bring up 0626 (=working cell)
0664	0668e90626e8	Add number in halo || Store result in 0626
0665	055ee30666e4	Go to right of 055E || Enter fixed point at 0666
0666	0bad82058be6	Error stop || Bring up "1"
0667	0668e80663e3	Store "1" in 0668 || Go to right side 0663
0668	000000000000	
0669	000000000000	
066a	000000000000	
066b	000000000000	
066c	05a3e605bfe9	Bring up 2m || Add "0" or "1" from 05BF
066d	0668e8065ee3	Store in 0668 || Go to right side of 065E

066e	05bfe605c1e8	Bring up "0" or "1" || Store "0" or "1" in 05C1	
066f	0589e605f8e8	Bring up upper limit of d_n constants || Store in 05F8	
0670	05fae80632e6	Store in 05FA || Bring up zero	
0671	0604e8058be6	Store "0" in 0604 || Bring up "1"	
0672	1E02E80593E6	Store "1" in 1E02 || Bring up $\ell-m$	
0673	058DEA0602E8	Subtract "2" || Store $\ell-m-2$ in 0602	
0674	05f8e6058dea	Bring up 05F8 || Subtract 2	
0675	05f8e80000e0	Store in limit cell (= 05F8) || *	
0676	0602e605f8ea	Bring up 0602 || Subtract (limit - 2x)	
0677	06a8f40000e0	If 0602 < (limit -2x) Go to 0678 (left side): If 0602 ≥ (limit -2x) Go to 06AB || *	
0678	05fae605c1e8		Calc. β
0679	067ae40000e0		
067a	067c50067b10	Final calc. of $N^m_{\ell-m+4}$ to N^m_k (k = max. subscript)	
067b	044420044422		
067c	067c50020010		
067d	0438e2067ee4		
067e	068050067f10		Calc. τ
067f	043720043722		
0680	068050020010		
0681	042ae205d1e6		

0682	05b9ea0604e9	
0683	0606e805e0e7	Calc. $N_k^m = \dfrac{-\beta_k^m}{-A_{m\ell} + j_k^m + N_{k+2}^m}$, where k = limit -2x
0684	0606ec0604e8	
0685	1e00e506b5e8	Store N_k^m in 1E00 || Store N_k^m in 06B5 (largest k to start)
0686	1002d0068500	
0687	05fae6058dea	|| Float left 0674
0688	05fae80674e2	Change exit of [N_n^m] routine to read "go to 068E".
0689	068EE20000E0	Bring up n = ℓ-m+2 || Add 2
068a	05fae6058de9	Store ℓ-m+4 in 05F8 || Bring up "0" or "1"
068b	05f8e805bfe6	Final calc. of N_o^m to $N_{\ell-m+2}^m$
068c	0602e80674e2	Store "0" or "1" in 0602 || Go to 0674
068d	068e00000000	
068e	0002d0068500	
068f	0685d8069000	
0690	0000e6058bdc	
0691	0692e20696e2	
0692	1902e80000e0	
0693	0002d0069200	
0694	0002d0069000	
0695	0690e2000000	
0696	0646d8069800	
0697	0636d9069900	
0698	0000d1069900	
0699	058be60000e8	
069a	0518e20000e0	Go to 0518 || *

069b	069fe6058be9	Bring up counter base (= zero) || Add one
069c	069fe8058fdc	Store count in 069F || Is this number greater than 3?
069d	0502e20502e2	Number less than 3. Go to 0502 || Number equal to 3: Go to 0502
069e	06a1e20000e0	Number greater than "3". Go to 06A1 || *
069f	000000000000	
06a0	000000000000	
06a1	05fee606a4ec	Bring up $10^{-2}/x$ || Divide by "2" (= 06A4)
06a2	05fee80632e6	Store $10^{-2}/x$ (x = 1, 2, 4, etc) in 05FE || Bring up zero
06a3	069fe8087ae2	Store in 069F || Go to 087A
06a4	080000000000	"2"
06a5	000200000000	
06a6	000000000000	
06a7	000000000000	
06a8	06b5e60000f9	
06a9	06b7e80000e0	
06aa	06b0d8067700	
06ab	06b1d8067200	Increase "n" in $(d_n)_{max}$ by 10. Then recompute
06ac	06b2d8068500	all N_k^m (k = limit -2x)
06ad	0589e606a4e9	
06ae	06b3e805f8e8	
06af	0670e20000e0	
06b0	06b900000000	
06b1	1e0200000000	
06b2	1e0000000000	
06b3	000000000000	
06b4	000000000000	
06b5	000000000000	
06b6	000000000000	
06b7	000000000000	
06b8	000000000000	

06b9	06b7e706b5fd	
06ba	0000f906c3dc	
06bb	0689e20689e2	
06bc	06b5e60000f9	
06bd	06b7e80000e0	If $[N_{\ell-m+2}]_1$ for n = max + 10 is not same (within tolerance) as $[N_{\ell-m+2}]_1$ for n = max recompute using a still larger n. If they agree to go to 0689
06be	06b1d8067200	
06bf	06b2d8068500	
06c0	06b3e606a4e9	
06c1	06b3e805f8e8	
06c2	0670e20000e0	
06c3	0afec0000000	
06c4	ffdc00000000	
06c5	05b9e60000f9	Bring up $A_{m\ell}$ || Form abs value
06c6	06d2e80000e0	Store in 06D2 || *
06c7	06c9d904ff00	Transfer left address of 06C0 to replace right address of 04FF
06c8	069be20000e0	Go to 069B || *
06c9	06ca00000000	
06ca	05b9e60000f9	Bring up $A_{m\ell}$ || Form abs value
06cb	06d2ea0000f9	Subtract previous $\|A_{m\ell}\|$ || Form abs value
06cc	06d0ea06cef4	Subtract tolerance in 06D0 || If $A_{m\ell}$ is not precise go to 06CE
06cd	064de2000000	If $A_{m\ell}$ meets precision go to 064D || *
06ce	05b9e60000f9	Bring up latest $A_{m\ell}$ || form abs value
06cf	06d2e8069be2	Store in 06D2 || Go to 069B
06d0	0afec0000000	
06d1	ffdc00000000	
06d2	000000000000	
06d3	000000000000	
06d4	0800000000f0	
06d5	000100000000	

06d6	07b3e6058bea	Bring up number from 07B3 ‖ Subtract 1.000. . . .
06d7	06dbf406d8e2	If zero go to 06DB ‖ If negative go to 06D8
06d8	0587e60000f9	Bring up h ‖ Absolute value of h
06d9	07dddc0420e2	Is h larger than m992? ‖ If h is less than m992 go to 0420
06da	06dee206dee2	If h equals m992 or is greater than m992 go to 06DE
06db	0587e60000f9	Bring up h ‖ Form absolute value of h
06dc	07dddc0420e2	Is h larger than m992? ‖ If h is less than m992 go to 0420
06dd	073de2073de2	If h equals m992 or is greater than m992 go to 073D
06de	0585e60583ea	
06df	058deb058be9	
06e0	07c7e807c7eb	
06e1	07c9e807c7eb	
06e2	07dfe807c7eb	
06e3	07e1e807c7eb	
06e4	07e3e807c7eb	
06e5	07e5e807c7eb	
06e6	07e7e80587e6	
06e7	0587eb07ece8	
06e8	0587eb07eee8	
06e9	0587eb07f0e8	
06ea	0587eb07f2e8	
06eb	0583e60583eb	
06ec	07f4e807f4eb	
06ed	0806e80000e0	lines 06DE to 0737
06ee	0587e607c7eb	compute all
06ef	07f6e807c9e6	terms in starting value of $A_{m\ell}$ for $h \to \infty$ Using Meixner (1) Eq. 10 (Prolate case)
06f0	0791e90793ec	
06f1	07f8e807f4e6	
06f2	0797eb07fae8	
06f3	07fae707c9e9	

06f4	0795e907c7eb
06f5	0799ec0587ec
06f6	07fce807c9e6
06f7	079beb07e1e9
06f8	079de90791eb
06f9	07fee807c9e6
06fa	058be907f4eb
06fb	07a1eb07fae8
06fc	07fae707fee9
06fd	079fec07ecec
06fe	0800e807e3e6
06ff	07a5eb07fae8
0700	07dfe607a7eb
0701	07fee807c7e6
0702	07a9eb07fee9
0703	07fae907a3ec
0704	07a3ec0802e8
0705	07c7e607adeb
0706	07fae807dfe6
0707	07abeb07fae9
0708	07f4eb07a3ec
0709	0804e807c7e6
070a	0806eb0793ec
070b	0804ea0802e9
070c	07eeec0808e8
070d	07e5e607b1eb
070e	07fae807e1e6
070f	07cbeb07fee8
0710	07c9e607cdeb
0711	07cfe907fee9

0712	07fae907afec
0713	07afec0804e8
0714	07e1e607d5eb
0715	07fae807c9e6
0716	07d1eb07fee8
0717	07d7e907fee9
0718	07fae907f4eb
0619	07d3ec0802e8
071a	07c9e6058be9
071b	0806eb07c5eb
071c	0793ec0802ea
071d	0804e907f0ec
071e	080ae807e7e6
071f	07d9eb07fae8
0720	07e3e607dbeb
0721	07fee807dfe6
0722	07b5eb0802e8
0723	07c7e607b7eb
0724	0802e907fee9
0725	07fae9079fec
0726	079fec07fae8
0727	07e3e607b9eb
0728	07fee807dfe6
0729	07bbeb0802e8
072a	07c7e607bdeb
072b	0802e907fee9
072c	07f4eb0797ec
072d	079fec07fee8
072e	07bfe607dfeb
072f	0802e807c7e6

0730	07c1eb0802e9	
0731	0806eb07d3ec	
0732	0802e807c7e6	
0733	07f4eb07c3ec	
0734	0804e80804e7	
0735	0802e907feea	
0736	07fae907f2ec	
0737	080ce80865e2	|| Go to 0865
0738	07f6e607f4e9	
0739	07f8ea07fcea	lines 0738 to 073C
073a	0800ea0808ea	compute starting $A_{m\ell}$ for $h \to \infty$ (Prolate case)
073b	080aea080cea	
073c	05b9e80420e2	|| Go to 0420
073d	0567e60587eb	
073e	0835e80587eb	
073f	0837e80587eb	
0740	0839e80587eb	
0741	083be80585e6	
0742	0583ea058deb	
0743	0583e9058be9	
0744	083de8083deb	
0745	083fe8083feb	Lines 073D to 078B compute all
0746	0841e8083feb	terms in starting value of $A_{m\ell}$ for $h \to \infty$
0747	0843e80583e6	Using Meixner (1) Eq. (19) (Oblate case)
0748	0583eb0845e8	
0749	0845eb0847e8	
074a	0845eb0849e8	
074b	083de60587eb	
074c	058deb084be8	
074d	083fe60845ea	

074e	058be9058dec
074f	084de80833ec
0750	0587ec083deb
0751	084fe80841e6
0752	0791eb0859e8
0753	083fe607ddeb
0754	085be8083fe6
0755	07c5eb058be9
0756	0845eb058deb
0757	085de8085de7
0758	0847e9058be9
0759	085be90859e9
075a	0799ec0835ec
075b	0851e80841e6
075c	07a5eb0859e8
075d	083fe60811eb
075e	085be8083fe6
075f	080feb0815e9
0760	0845eb058deb
0761	085de80847e6
0762	0817eb085dea
0763	07abe9085be9
0764	0859e9083deb
0765	07d3ec0837ec
0766	0853e80843e6
0767	07b1eb0859e8
0768	0841e60813eb
0769	085be8083fe6
076a	0819eb085de8
076b	0841e607ddeb

076c	085fe8083fe6
076d	080feb07c5e9
076e	085fe90845eb
076f	07ddeb085fe8
0770	083fe60817eb
0771	081de90847eb
0772	07c5eb0861e8
0773	0849e6058deb
0774	0863e80863e7
0775	0861e9085fea
0776	081be9085de9
0777	085be90859e9
0778	079fec0839ec
0779	0855e80843e6
077a	07d9eb0859e8
077b	0841e6081feb
077c	085be8083fe6
077d	0821eb085de8
077e	0841e60825eb
077f	085fe8083fe6
0780	0827eb0829e9
0781	085fe90845eb
0782	085fe8083fe6
0783	082beb082de9
0784	0847eb0861e8
0785	0849e6082feb
0786	0863e80863e7
0787	0861e9085fea
0788	0823e9085de9
0789	085be90859e9

078a	083deb0831ec	
078b	083bec086de2	‖ Go to 086D
078c	0835e7084be9	
078d	084dea084fea	lines 078C to 0790
078e	0851ea0853ea	compute starting value of $A_{m\ell}$ for $h \to \infty$ (Oblate case)
078f	0855ea0857ea	
0790	05b9e80420e2	‖ Go to 0420
0791	0a0000000000	"5"
0792	000300000000	
0793	080000000000	
0794	000400000000	
0795	0b0000000000	
0796	000400000000	
0797	080000000000	
0798	000600000000	
0799	080000000000	
079a	000700000000	
079b	0d0000000000	
079c	000500000000	
079d	0a8000000000	
079e	000500000000	
079f	080000000000	
07a0	000b00000000	
07a1	0c0000000000	
07a2	000900000000	
07a3	080000000000	
07a4	000800000000	
07a5	084000000000	
07a6	000600000000	
07a7	0c7400000000	

07a8	000b00000000
07a9	0afa80000000
07aa	000d00000000
07ab	094000000000
07ac	000600000000
07ad	0a7000000000
07ae	000800000000
07af	080000000000
07b0	000900000000
07b1	0fc000000000
07b2	000600000000
07b3	000000000000
07b4	000000000000
07b5	0fedf9000000
07b6	001400000000
07b7	088d0fc00000
07b8	001600000000
07b9	0b3580000000
07ba	000d00000000
07bb	0f91f0000000
07bc	001100000000
07bd	091f8e000000
07be	001300000000
07bf	0b1800000000
07c0	000900000000
07c1	0bc200000000
07c2	000f00000000
07c3	080000000000
07c4	000500000000
07c5	0c0000000000

07c6	000200000000
07c7	000000000000
07c8	000000000000
07c9	000000000000
07ca	000000000000
07cb	09a600000000
07cc	000d00000000
07cd	0a93f0000000
07ce	001000000000
07cf	0af8c0000000
07d0	000f00000000
07d1	0a3c00000000
07d2	000b00000000
07d3	030000000000
07d4	000a00000000
07d5	0e6000000000
07d6	000700000000
07d7	0b7c00000000
07d8	000a00000000
07d9	083c00000000
07da	000a00000000
07db	0f0590000000
07dc	001000000000
07dd	000000000000
07de	000000000000
07df	000000000000
07e0	000000000000
07e1	000000000000
07e2	000000000000
07e3	000000000000

07e4	000000000000
07e5	000000000000
07e6	000000000000
07c7	000000000000
07e8	000000000000
07e9	00000007ea10
07ca	000000000000
07eb	000000000000
07ec	000000000000
07ed	000000000000
07ce	000000000000
07cf	000000000000
07f0	000000000000
07f1	000000000000
07f2	000000000000
07f3	000000000000
07f4	000000000000
07f5	000000000000
07f6	000000000000
07f7	000000000000
07f8	000000000000
07f9	000000000000
07fa	000000000000
07fb	000000000000
07fc	000000000000
07fd	000000000000
07fe	000000000000
07ff	000000000000
0800	000000000000
0801	000000000000

0802	000000000000
0803	000000000000
0804	000000000000
0805	000000000000
0806	000000000000
0807	000000000000
0808	000000000000
0809	000000000000
080a	000000000000
080b	000000000000
080c	000000000000
080d	000000000000
080e	000000080f10
080f	0b8000000000
0810	000500000000
0811	0e4000000000
0812	000700000000
0813	0aa000000000
0814	000900000000
0815	0c8000000000
0816	000500000000
0817	0d0000000000
0818	000400000000
0819	0ef000000000
081a	000800000000
081b	0e0000000000
081c	000400000000
081d	0c0000000000
081e	000300000000
081f	081580000000

0820	000d00000000
0821	0a3280000000
0822	001100000000
0823	0fc400000000
0824	000a00000000
0825	0eac00000000
0826	000a00000000
0827	0ea600000000
0828	000c00000000
0829	0c6e00000000
082a	000b00000000
082b	0e8800000000
082c	000900000000
082d	09ec00000000
082e	000a00000000
082f	0d4000000000
0830	000600000000
0831	080000000000
0832	000e00000000
0833	080000000000
0834	000300000000
0835	000000000000
0836	000000000000
0837	000000000000
0838	000000000000
0839	000000000000
083a	000000000000
083b	000000000000
083c	000000000000
083d	000000000000

083e	000000000000
083f	000000000000
0840	000000000000
0841	000000000000
0842	000000000000
0843	000000000000
0844	000000000000
0845	000000000000
0846	000000000000
0847	000000000000
0848	000000000000
0849	000000000000
084a	000000000000
084b	000000000000
084c	000000000000
084d	000000000000
084e	000000000000
084f	000000000000
0850	000000000000
0851	000000000000
0852	000000000000
0853	000000000000
0854	000000000000
0855	000000000000
0856	000000000000
0857	000000000000
0858	000000000000
0859	000000000000
085a	000000000000
085b	000000000000

085c	000000000000	
085d	000000000000	
085e	000000000000	
085f	000000000000	
0860	000000000000	
0861	000000000000	
0862	000000000000	
0863	000000000000	
0864	000000000000	
865	07f6e60000f9	Bring up 1st term in Meixner Formula (Prolate case) || Form abs value
0866	0835e8080ce6	Store in 0835 || Bring up last term in Meixner Formula
0867	0000f90837e8	Form abs value || Store in 0837
0868	08t5e6086beb	Bring up 0835 || Multiply by $10^{-2}/2$
0869	0837ea0738f4	Subtract 0837 || If positive go to 0738; If neg. go to 086A
086a	0420e20000e0	Go to 0420 || *
086b	0a3d70a3d70a	$10^{-2}/2$
086c	fff900000000	
086d	0857e80835e6	Store in 0857 || Bring up 1st term in Meixner Formula (Oblate case)
086e	0000f907f6e8	Form abs value || Store in 07F6
086f	0857e60000f9	Bring up 0857 (= last term in Meixner Formula) || Form abs value
0870	07f8e807f6e6	Store in 07F8 || Bring up 07F6
0871	086beb07f8ea	Multiply by $10^{-2}/2$ || Subtract 07F8
0872	078cf40420e2	If positive go to 078C; If negative go to right side || Go to 0420
0873	0025f30010f3	Carriage return || Space (in type out)
0874	0010f30010f3	Space || Space
0875	0010f30029f3	Space ||
0876	0610e6000bf0	Bring up 0610 (= tol on eigenvalue) || Print 11
0877	0501e40000e0	Enter fixed point at left side of 0501 || *
0878	0000e00000e0	* || *
0879	0000e00000e0	* || *

087a	0610e60791eb	Bring up Tol of Eigenvalue || Multiply by 5
087b	058deb0610e8	Multiply by 2 || Store in 0610
087c	06d0e80000e0	Store in 06D0 || *
087d	051ed9069c00	Place the left address of 051E into the right address of 069C
087e	0502e20000e0	Go to 0502 || *
087f	046ed80692 00	Place the left address of 046E into left address of 0692
0880	1002d0069200	Decrement left address of 0692 by 2
0881	0467c20000c0	Go to 0467 || *

mmmmmmm

UNCLASSIFIED

Naval Research Laboratory. Report 5837.
A DIGITAL COMPUTER PROGRAM FOR CALCULATING SPHEROIDAL WAVE FUNCTION EIGENVALUES AND EXPANSION CONSTANTS, by S. Hanish and R.V. Baier. 69 pp. and figs., January 31, 1963.

Using the well-known procedures of Bouwkamp ("Theoretische en numerieke behandling van de buiging door een ronde opening," Diss. Groningen, Groningen Batavia, 1941) for finding the eigenvalues of differential equations solvable by application of three-term recursion formulas, the authors have developed a digital computer program in symbolic language for obtaining the eigenvalues of the separated steady-state wave equation in spheroidal coordinates, i.e., for the equation

UNCLASSIFIED (Over)

1. Mathematical computer programming
2. Wave functions – Analysis
I. Hanish, S.
II. Baier, R. V.

UNCLASSIFIED

Naval Research Laboratory. Report 5837.
A DIGITAL COMPUTER PROGRAM FOR CALCULATING SPHEROIDAL WAVE FUNCTION EIGENVALUES AND EXPANSION CONSTANTS, by S. Hanish and R.V. Baier. 69 pp. and figs., January 31, 1963.

Using the well-known procedures of Bouwkamp ("Theoretische en numerieke behandling van de buiging door een ronde opening," Diss. Groningen, Groningen Batavia, 1941) for finding the eigenvalues of differential equations solvable by application of three-term recursion formulas, the authors have developed a digital computer program in symbolic language for obtaining the eigenvalues of the separated steady-state wave equation in spheroidal coordinates, i.e., for the equation

UNCLASSIFIED (Over)

1. Mathematical computer programming
2. Wave functions – Analysis
I. Hanish, S.
II. Baier, R. V.

UNCLASSIFIED

Naval Research Laboratory. Report 5837.
A DIGITAL COMPUTER PROGRAM FOR CALCULATING SPHEROIDAL WAVE FUNCTION EIGENVALUES AND EXPANSION CONSTANTS, by S. Hanish and R.V. Baier. 69 pp. and figs., January 31, 1963.

Using the well-known procedures of Bouwkamp ("Theoretische en numerieke behandling van de buiging door een ronde opening," Diss. Groningen, Groningen Batavia, 1941) for finding the eigenvalues of differential equations solvable by application of three-term recursion formulas, the authors have developed a digital computer program in symbolic language for obtaining the eigenvalues of the separated steady-state wave equation in spheroidal coordinates, i.e., for the equation

UNCLASSIFIED (Over)

1. Mathematical computer programming
2. Wave functions – Analysis
I. Hanish, S.
II. Baier, R. V.

UNCLASSIFIED

Naval Research Laboratory. Report 5837.
A DIGITAL COMPUTER PROGRAM FOR CALCULATING SPHEROIDAL WAVE FUNCTION EIGENVALUES AND EXPANSION CONSTANTS, by S. Hanish and R.V. Baier. 69 pp. and figs., January 31, 1963.

Using the well-known procedures of Bouwkamp ("Theoretische en numerieke behandling van de buiging door een ronde opening," Diss. Groningen, Groningen Batavia, 1941) for finding the eigenvalues of differential equations solvable by application of three-term recursion formulas, the authors have developed a digital computer program in symbolic language for obtaining the eigenvalues of the separated steady-state wave equation in spheroidal coordinates, i.e., for the equation

UNCLASSIFIED (Over)

1. Mathematical computer programming
2. Wave functions – Analysis
I. Hanish, S.
II. Baier, R. V.

UNCLASSIFIED

$$\frac{d}{d\eta}\left[(1-\eta^2)\frac{dS_{m\ell}}{d\eta}\right]+\left[A_{m\ell}-h^2\eta^2-\frac{m^2}{1-\eta^2}\right]S_{m\ell}=0$$

where the parameters ℓ, m, and h are as defined by Stratton et al. in "Spheroidal Wave Functions" (New York:Wiley, 1956). The symbolic language used is universal (e.g., add a to b, multiply c by d, etc.) and may be applied to any digital computer by simple conversion to machine language. The program developed requires at least 3400 working cells. It computes the eigenvalue $A_{m\ell}$ to 13 significant places for any integer values of ℓ and m ($\ell \geq m$). The value of h is left arbitrary. The program also computes all the spheroidal wave function expansion coefficients (d_p) up to $p \leq 200$.

UNCLASSIFIED

UNCLASSIFIED

$$\frac{d}{d\eta}\left[(1-\eta^2)\frac{dS_{m\ell}}{d\eta}\right]+\left[A_{m\ell}-h^2\eta^2-\frac{m^2}{1-\eta^2}\right]S_{m\ell}=0$$

where the parameters ℓ, m, and h are as defined by Stratton et al. in "Spheroidal Wave Functions" (New York:Wiley, 1956). The symbolic language used is universal (e.g., add a to b, multiply c by d, etc.) and may be applied to any digital computer by simple conversion to machine language. The program developed requires at least 3400 working cells. It computes the eigenvalue $A_{m\ell}$ to 13 significant places for any integer values of ℓ and m ($\ell \geq m$). The value of h is left arbitrary. The program also computes all the spheroidal wave function expansion coefficients (d_p) up to $p \leq 200$.

UNCLASSIFIED

UNCLASSIFIED

$$\frac{d}{d\eta}\left[(1-\eta^2)\frac{dS_{m\ell}}{d\eta}\right]+\left[A_{m\ell}-h^2\eta^2-\frac{m^2}{1-\eta^2}\right]S_{m\ell}=0$$

where the parameters ℓ, m, and h are as defined by Stratton et al. in "Spheroidal Wave Functions" (New York:Wiley, 1956). The symbolic language used is universal (e.g., add a to b, multiply c by d, etc.) and may be applied to any digital computer by simple conversion to machine language. The program developed requires at least 3400 working cells. It computes the eigenvalue $A_{m\ell}$ to 13 significant places for any integer values of ℓ and m ($\ell \geq m$). The value of h is left arbitrary. The program also computes all the spheroidal wave function expansion coefficients (d_p) up to $p \leq 200$.

UNCLASSIFIED

UNCLASSIFIED

$$\frac{d}{d\eta}\left[(1-\eta^2)\frac{dS_{m\ell}}{d\eta}\right]+\left[A_{m\ell}-h^2\eta^2-\frac{m^2}{1-\eta^2}\right]S_{m\ell}=0$$

where the parameters ℓ, m, and h are as defined by Stratton et al. in "Spheroidal Wave Functions" (New York:Wiley, 1956). The symbolic language used is universal (e.g., add a to b, multiply c by d, etc.) and may be applied to any digital computer by simple conversion to machine language. The program developed requires at least 3400 working cells. It computes the eigenvalue $A_{m\ell}$ to 13 significant places for any integer values of ℓ and m ($\ell \geq m$). The value of h is left arbitrary. The program also computes all the spheroidal wave function expansion coefficients (d_p) up to $p \leq 200$.

UNCLASSIFIED

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.